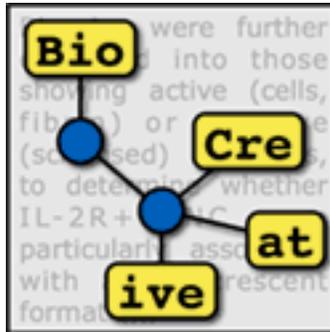


# Proceedings of the Fourth BioCreative Challenge Evaluation Workshop vol. 1



October 7-9, 2013  
Bethesda, MD USA

## Editors:

- Cecilia N. Arighi
- Kevin B. Cohen
- Lynette Hirschman
- Zhiyong Lu
- Catalina O. Tudor
- Thomas Wieggers
- W. John Wilbur
- Cathy H. Wu

ISBN 978-0-615-89815-5

# Table of Contents

<b>Preface</b>	<b>v</b>
<b>Committees</b>	<b>vi</b>
<b>Workshop Agenda</b>	<b>vii</b>
<b>TRACK 1 (BioC: Interoperability)</b>	<b>1</b>
<b>PyBioC: a python implementation of the BioC core</b>	<b>2</b>
Hernani Marques, Fabio Rinaldi	
<b>Enhancing the Interoperability of iSimp by Using the BioC Format</b>	<b>5</b>
Yifan Peng, Catalina O Tudor, Manabu Torii, Cathy H Wu, K Vijay-Shanker	
<b>Improving Interoperability of Text Mining Tools with BioC</b>	<b>10</b>
Ritu Khare, Chih-Hsuan Wei, Yuqing Mao, Robert Leaman, Zhiyong Lu	
<b>Finding Abbreviations in Biomedical Literature: Three BioC-Compatible Modules and Three BioC-formatted Corpora</b>	<b>23</b>
Rezarta Islamaj Doğan, Donald C. Comeau, Lana Yeganova and W. John Wilbur	
<b>Extending BioC Implementation to More Languages</b>	<b>31</b>
Wanli Liu, Donald C. Comeau, Rezarta Islamaj Doğan, and W. John Wilbur	
<b>Natural Language Processing Pipelines to Annotate BioC Collections with an Application to the NCBI Disease Corpus</b>	<b>38</b>
Donald C. Comeau, Haibin Liu, Rezarta Islamaj Doğan and W. John Wilbur	
<b>Brat2BioC: conversion tool between brat and BioC</b>	<b>46</b>
Antonio Jimeno Yepes, Mariana Neves, Karin Verspoor	
<b>A Biomedical Semantic Role Labeling BioC Module for BioCreative IV</b>	<b>54</b>
Po-Ting, Lai, Hong-Jie Dai, Johnny Chi-Yang Wu and Richard Tzong-Han Tsai	
<b>NaCTeM's BioC Modules and Resources for BioCreative IV</b>	<b>61</b>
Rafal Rak, Riza Batista-Navarro, Andrew Rowley, Makoto Miwa, Jacob Carter and Sophia Ananiadou	
<b>TRACK 3 (CTD)</b>	<b>68</b>
<b>Web services-based text mining demonstrates broad impacts for interoperability and process simplification</b>	<b>69</b>
Thomas C. Wieggers, Allan Peter Davis, and Carolyn J. Mattingly	
<b>NaCTeM CTD Web Services</b>	<b>85</b>
Riza Batista-Navarro, Rafal Rak and Sophia Ananiadou	
<b>OntoGene: CTD entity and action term recognition</b>	<b>90</b>
Fabio Rinaldi, Simon Clematide, Tilia Renate Ellendorff, Hernani Marques	
<b>Performance of a multi-class biomedical tagger on the BioCreative IV CTD task</b>	<b>95</b>
S. V. Ramanan and P. Senthil Nathan	
<b>A Web Service Annotation Framework for CTD Using the UIMA Concept Mapper</b>	<b>99</b>
Andrew MacKinlay and Karin Verspoor	
<b>Multi-stage Gene Mention Identification Method for BioCreative IV Track 3</b>	<b>103</b>
Po-Ting, Lai, Kuan-Chieh, Chung, Johnny Chi-Yang Wu, Hong-Jie Dai, and Richard Tzong-Han Tsai	
<b>ToxiCat: Hybrid Named Entity Recognition services to support curation of the Comparative Toxicogenomic Database</b>	<b>108</b>
Dina Vishnyakova, Julien Gobeill, Emilie Pasche and Patrick Ruch	
<b>Adapting the OCMiner text processing system to the CTD controlled vocabulary</b>	<b>114</b>
Matthias Irmer, Claudia Bobach, Timo Böhme, Ulf Laube, Anett Püschel, Lutz Weber	

<b>TRACK 4 (GO)</b>	<b>118</b>
<b>The Gene Ontology Task at BioCreative IV</b>	<b>119</b>
Yuqing Mao, Kimberly Van Auken, Donghui Li, Cecilia N. Arighi, Zhiyong Lu	
<b>Corpus Construction for the BioCreative IV GO Task</b>	<b>128</b>
Kimberly Van Auken, Mary L. Schaeffer, Peter McQuilton, Stanley J. F. Laulederkind, Donghui Li, Shur-Jen Wang, G. Thomas Hayman, Susan Tweedie, Cecilia N. Arighi, James Done, Hans-Michael Müller, Paul W. Sternberg, Yuqing Mao, Chih-Hsuan Wei and Zhiyong Lu	
<b>BiTeM/SIBtex group proceedings for BioCreative IV, Track 4: Gene Ontology curation</b>	<b>139</b>
Gobeill Julien, Pasche Emilie, Vishnyakova Dina and Ruch Patrick	
<b>Integrating Information Retrieval with Distant Supervision for Gene Ontology Annotation</b>	<b>146</b>
Dongqing Zhu, Dingcheng Li, Ben Carterette, Hongfang Liu	
<b>Unsupervised Information Extraction for Finding Gene Functions</b>	<b>156</b>
Ehsan Emadzadeh, Azadeh Nikfarjam, Rachel E. Ginn and Dr. Graciela Gonzalez	
<b>A Robust Data-Driven Approach for BioCreative IV GO Annotation Task</b>	<b>162</b>
Yanpeng Li, Abhyuday Jagannat and Hong Yu	
<b>Gene Ontology Evidence Sentence Retrieval Using Combinatorial Applications of Semantic Class and Rule Patterns</b>	<b>169</b>
Jian-Ming Chen, Yung-Chun Chang, Johnny Chi-Yang Wu, Po-Ting Lai, Hong-Jie Dai	
<b>Gene Ontology Concept Recognition using Cross-Products and Statistical Methods</b>	<b>174</b>
Luu Anh Tuan, Jung-jae Kim, See-Kiong Ng	
<b>Gene Ontology Evidence Sentence Extraction and Concept Extraction: Two Rule-Based Approaches</b>	<b>182</b>
Yu-De Chen, Chia-Jung Yang, Wen-Gan Li, Chin-Yu Huang, Jung-Hsien Chiang	
<b>TRACK 5 (IAT)</b>	<b>189</b>
<b>BioCreative IV Interactive Task</b>	<b>190</b>
Sherri Matis-Mitchell, Phoebe Roberts, Catalina O. Tudor and Cecilia N. Arighi	
<b>Evaluation of the CellFinder pipeline in the BioCreative IV User Interactive task</b>	<b>204</b>
<b>Assisted curation of growth conditions that affect gene expression in <i>E. coli</i> K-12</b>	<b>214</b>
Socorro Gama-Castro, Fabio Rinaldi, Alejandra López-Fuentes, Yalbi Itzel Balderas-Martínez, Simon Clematide, Tilia Renate Ellendorff, Julio Collado-Vides	
<b>ODIN: a customizable literature curation tool</b>	<b>219</b>
Fabio Rinaldi, Allan Peter Davis, Christopher Southan, Simon Clematide, Tilia Renate Ellendorff, Gerold Schneider	
<b>MarkerRIF: An Interactive Curation System for Biomarker</b>	<b>224</b>
Hong-Jie Dai, Chi-Yang Wu, Wei-San Lin, Richard Tzong-Han Tsai, Wen-Lian Hsu	
<b>Supporting Document Triage with the SciKnowMine System in the Mouse Genome Informatics (MGI) Curation Process</b>	<b>234</b>
Gully APC Burns, Marcelo Tallis, Hiroaki Onda, Kevin Cohen, James Kadin, Judith Blake	
<b>BioQRator: a web-based interactive biomedical literature curating system</b>	<b>241</b>
Dongseop Kwon, Sun Kim, Soo-Yong Shin, and W. John Wilbur	
<b>RLIMS-P: Literature-based curation of protein phosphorylation information</b>	<b>247</b>
Manabu Torii, Gang Li, Zhiwen Li, Irem Çelen, Francesca Diella, Rose Oughtred, Cecilia Arighi, Hongzhan Huang, K. Vijay-Shanker, Cathy H. Wu	
<b>Egas – Collaborative Biomedical Annotation as a Service</b>	<b>254</b>
David Campos, Joni Lourenço, Tiago Nunes, Rui Vitorino, Pedro Domingues, Sérgio Matos and José Luís Oliveira	
<b>tagtog: Interactive Human and Machine Annotation of Gene Mentions in PLOS Full-Text Articles</b>	<b>260</b>
Juan Miguel Cejuela, Peter McQuilton, Laura Ponting, Steven J. Marygold, Raymund Stefancsik, Gillian H. Millburn, Burkhard Rost and the FlyBase Consortium	

<b>Customisable Curation Workflows in Argo</b>	<b>270</b>
Rafal Rak, Riza Batista-Navarro, Andrew Rowley, Jacob Carter and Sophia Ananiadou	
<b>METAGENOMICS PANEL</b>	<b>279</b>
<b>Overview</b>	<b>280</b>
<b>Evangelos Pafilis, PhD</b>	<b>282</b>
<b>James R. Cole, Ph.D.</b>	<b>286</b>
<b>George M. Garrity, Sc.D.</b>	<b>288</b>
<b>Folker Meyer</b>	<b>290</b>
<b>Tatiana Tatusova</b>	<b>291</b>
<b>POSTER ABSTRACTS</b>	<b>292</b>
<b>Towards automatic large-scale curation of genomic variation: improving coverage based on supplementary material</b>	<b>293</b>
Antonio Jimeno and Karin Verspoor	
<b>WBI resources and participation in BioCreative IV</b>	<b>295</b>
Torsten Huber, Mariana Neves, Tim Rocktäsc, Philippe Thomas, Michael Weidlich, and Ulf Leser	
<b>Co-occurrence Interaction Nexus with Named-entity Recognition</b>	<b>297</b>
Yi-Yu Hsu and Hung-Yu Kao	
<b>tmVar: A new machine learning method for mutation extraction in biomedical text</b>	<b>298</b>
Chih-Hsuan Wei, Bethany Harris, Hung-Yu Kao, and Zhiyong Lu	
<b>DNorm: A New Method and Tool for Disease Name Normalization</b>	<b>300</b>
Robert Leaman, Rezarta Islamaj Dogan, Ritu Khare, Chih-Hsuan Wei, and Zhiyong Lu	
<b>Mining the Impact of Phosphorylation on PPI in Full Length Scientific Articles</b>	<b>301</b>
Catalina O. Tudor, Cecilia N. Arighi, Cathy H. Wu, K. Vijay-Shanker	

# Preface

Welcome **to the Fourth BioCreative workshop** being held in **Washington DC, USA on October 7-9, 2013**. On behalf of the Organizing Committee, we would like to thank you for your participation and hope you enjoy the workshop.

The BioCreative (Critical Assessment of Information Extraction systems in Biology) challenge evaluation consists of a community-wide effort for evaluating text mining and information extraction systems applied to the biological domain (<http://www.biocreative.org/>). Its aim is to promote the development of text mining and text processing tools which are useful to the communities of researchers and database curators in the biological sciences. The main emphasis is on the comparison of methods and the community assessment of scientific progress, rather than on the purely competitive aspects.

BioCreative I (2004) focused on the extraction of gene or protein names from text, and their mapping into standardized gene identifiers (GN) for three model organism databases, and functional annotation, requiring systems to identify specific text passages that supported Gene Ontology annotations for specific proteins, given full text articles.

BioCreative II (2007) focused on the GN task but for human genes or gene products mentioned in PubMed/MEDLINE abstracts, and on protein-protein interaction (PPI) extraction, based on the main steps of a manual protein interaction annotation workflow.

BioCreative II.5 (2009) focused on the PPI task: ranking of articles for curation purposes, and identifying the interacting proteins in the positive articles.

The BioCreative III (2010) continued the tradition of a challenge evaluation on several tasks, including a gene normalization (GN) task and two protein-protein interaction (PPI) tasks (PPI article classification, and PPI method detection). It also introduced a new interactive task (IAT), ran as a demonstration task. The goal of IAT was to develop an interactive system to facilitate a user's annotation of the unique database identifiers for all the genes appearing in an article. This task included ranking genes by importance, taking into consideration the amount of experimental information described in the articles.

The BioCreative-2012 Workshop on Interactive Text Mining in the Biocuration Workflow aimed to bring together the biocuration and text mining communities towards the development and evaluation of interactive text mining tools and systems to improve utility and usability in the biocuration workflow. To achieve this goal, the workshop consisted of three Tracks: I-Triage, a collaborative biocuration-text mining development task for document prioritization for curation; II-Workflow, a biocuration workflow survey and analysis task; and III-Interactive TM, an interactive text mining and user evaluation task.

The BioCreative IV Workshop (2013) consists of five tracks: (1) BioC: Interoperability, for the development of an interoperable BioNLP module that can be seamlessly coupled with BioC compliant modules; (2) Chemical and Drug Named Entity Recognition (ChemDNER), for the detection of mentions of chemical compounds and drugs, in particular those chemical entity mentions that can subsequently be linked to a chemical structure; (3) Comparative Toxicogenomics Database (CTD) Curation, for provision of web services to identify gene, chemical, disease, and action term mentions supporting CTD curation in PubMed abstracts; (4) Gene Ontology (GO) curation, for development of automatic methods to aid GO curators in identifying articles with curatable GO information (triage) and extracting gene function terms and the associated evidence sentences in full-length articles; and (5) Interactive Curation (IAT), for demonstration and evaluation of web-based systems addressing user-defined tasks, evaluated by curators on performance and usability. This is the first volume of the proceedings, and includes Tracks 1, 3, 4, and 5.

We would like to thank all participating teams, panelists, and all the chairs and committee members. The BioCreative IV Workshop was supported by NSF grant DBI-0850319.

# Committees

## **Steering Committee**

Cecilia N. Arighi, University of Delaware, USA

Kevin B. Cohen, University of Colorado, USA

Lynette Hirschman, MITRE Corporation, USA

Martin Krallinger, Spanish National Cancer Centre (CNIO), Spain

Zhiyong Lu, National Center for Biotechnology Information (NCBI), NIH, USA

Catalina O. Tudor, University of Delaware, USA

Alfonso Valencia, Spanish National Cancer Centre (CNIO), Spain

Thomas Wieggers, North Carolina State University, USA

W. John Wilbur, National Center for Biotechnology Information (NCBI), NIH, USA

Cathy H. Wu, University of Delaware, USA

## **Local Organizing Committee**

Cecilia N. Arighi, University of Delaware, USA

Susan Phipps, University of Delaware, USA

Catalina O. Tudor, University of Delaware, USA

W. John Wilbur, National Center for Biotechnology Information (NCBI), NIH, USA

Cathy H. Wu, University of Delaware, USA

## **User Advisory Committee**

Judith Blake, MGI, The Jackson Laboratory, USA

Andrew Chatr-aryamontri, BioGrid, Canada

Stan Laulederkind, Rat Genome Database, USA

Donghui Li, TAIR, USA

Sherri Matis, Astrazeneca, USA

Fiona McCarthy, Agbase, USA

Peter McQuilton, Flybase, UK

Sandra Orchard, IntAct, UK

Phoebe Roberts, Pfizer, USA

Mary Schaeffer, MaizeGDB, USA

Kimberly Van-Auken, Wormbase, USA

## **Proceedings Committee**

Cecilia N. Arighi, University of Delaware, USA

Catalina O. Tudor, University of Delaware, USA

# Workshop Agenda

**Monday, October 7, 2013**

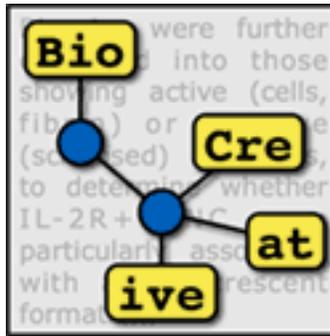
08:00 - 12:00	Registration
09:00 - 09:15	Workshop opening (W.J. Wilbur/C. Arighi)
09:15 - 11:15	<p><b>TRACK 1 BioC: Interoperability</b></p> <ul style="list-style-type: none"> <li>• 9:15 Overview of BioC and BioCreative IV - Interoperability track (D.C. Comeau/K.B. Cohen)</li> <li>• 9:45 BioC for NER Web Services-Based Inter-process Communications (T. Wiegers)</li> <li>• 10:00 PyBioC: a python implementation of the BioC core (F. Rinaldi)</li> <li>• 10:15 Enhancing the Interoperability of iSimp by Using the BioC Format (Y. Peng)</li> <li>• 10:30 Improving Interoperability of Text Mining Tools with BioC (C-H Wei/K.B. Cohen)</li> <li>• 10:45 Addressing common challenges in biomedical text processing with BioC: some processing tools and corpora (R.I. Dogan/W. Liu)</li> <li>• 11:00 Questions, Discussion and Conclusion (D.C. Comeau/K.B. Cohen and R.I. Dogan/W. Liu)</li> </ul>
11:15 - 11:35	Break
11:35 - 12:35	<p><b>TRACK 3 CTD (session 1)</b></p> <ul style="list-style-type: none"> <li>• 11:35 Overview and results of the BioCreative IV – CTD track (T. Wiegers)</li> <li>• 12:05 NaCTeM CTD Web Services (R. Batista-Navarro)</li> <li>• 12:20 Morning Discussion</li> </ul>
12:35 - 13:40	Lunch
13:40 - 14:40	<p><b>TRACK 3 CTD (session 2)</b></p> <ul style="list-style-type: none"> <li>• 13:40 OntoGene: CTD entity and action term recognition (F. Rinaldi)</li> <li>• 13:55 Adapting a multi-class biomedical tagger for the CTD task (S.V. Ramanan)</li> <li>• 14:10 A Web Service Annotation Framework for CTD Using the UIMA Concept Mapper (K. Verspoor)</li> <li>• 14:25 Afternoon Discussion</li> </ul>
14:40 - 15:00	Break
15:00 - 17:00	<p><b>TRACK 4 GO</b></p> <ul style="list-style-type: none"> <li>• 15:00 Overview and results of the BioCreative IV – GO track (Z. Lu/K. Van-Auken)</li> <li>• 15:30 Closing the loop: from paper to protein annotation using automatic text categorization (J. Gobeill)</li> <li>• 15:45 Integrating Information Retrieval with Distant Supervision for Gene Ontology Annotation (D. Zhu)</li> <li>• 16:00 Unsupervised Information Extraction for Finding Gene Functions (E. Emadzadeh)</li> <li>• 16:15 A Robust Data-Driven Approach for Gene Ontology Annotation (Y. Li)</li> <li>• 16:30 Discussion</li> </ul>
17:00 - 19:00	<p>POSTER/DEMO SESSION I and RECEPTION (Harmony Room)</p> <ul style="list-style-type: none"> <li>• Posters and Demos for Interoperability, GO, and CTD tracks</li> </ul>

## Tuesday, October 8, 2013

08:00 - 12:00	Registration
09:00 - 09:45	<b>Plenary Talk: Evangelos Pafilis</b> , Institute of Marine Biology Biotechnology and Aquaculture Hellenic Centre for Marine Research (HCMR), Heraklion, Crete, Greece SPECIES and ENVIRONMENTS: taxonomic name and environment descriptive term identification in text
09:45 - 11:15	<b>DOE PANEL (moderator: L. Hirschman)</b> Panelists: <ul style="list-style-type: none"> <li>• Jim Cole, Michigan State University</li> <li>• George Garrity, Names for Life Folker Meyer, Argonne National Laboratory</li> <li>• Nikos Kyrpides, Joint Genome Institute</li> <li>• Tatiana Tatusova, NCBI</li> </ul>
11:15 - 11:30	Break
11:30 - 12:45	<b>TRACK 2 CHEMDNER (session 1)</b> <ul style="list-style-type: none"> <li>• 11:30 Overview and results of the BioCreative IV – CHEMDNER task (M. Krallinger)</li> <li>• 12:00 PA dictionary- and grammar-based name entity recognizer for chemical name entity recognition (E. van Mulligen)</li> <li>• 12:15 tmChem: a machine-learning approach for recognizing chemical names in PubMed articles (R. Leaman/K. Verspoor)</li> <li>• 12:30 Extended Feature Set for Chemical Named Entity Recognition and Indexing (M. Neves)</li> </ul>
12:45 - 13:45	Lunch
13:45 - 15:45	<b>TRACK 2 CHEMDNER (session 2)</b> <ul style="list-style-type: none"> <li>• 13:45 Chemistry-specific Features and Heuristics for Developing a CRF-based Chemical Named Entity Recogniser (R. Navarro-Batista)</li> <li>• 14:00 In grammars we trust: LeadMine, a knowledge driven solution (D. Lowe)</li> <li>• 14:15 Chemical name recognition with harmonized feature-rich conditional random fields (D. Campos)</li> <li>• 14:30 Recognizing Chemical Named Entities using CRF and SSVM (H. Xu)</li> <li>• 14:45 Adapting Cocoa, a multi-class entity detector, for the CHEMDNER task of BioCreative IV ( S.V. Ramanan)</li> <li>• 15:00 TBD (T. Munkhdalai)</li> <li>• 15:15 CHEMDNER discussion: organizers &amp; all participating teams</li> </ul>
15:45 - 16:00	Break
16:00 - 18:00	POSTER/DEMO SESSION II (Harmony Room) <ul style="list-style-type: none"> <li>• Posters and Demos for ChemDNER track and other posters</li> </ul>

## Wednesday, October 9, 2013

09:00 - 11:00	<p><b>TRACK 5 IAT</b></p> <ul style="list-style-type: none"> <li>• 9:00 Overview and results of the BioCreative IV – (C. Arighi)</li> <li>• 9:20 Evaluation of the CellFinder pipeline in the BioCreative IV User Interactive task (M. Neves)</li> <li>• 9:30 Ontogene curation pipelines (F. Rinaldi)</li> <li>• 9:40 MarkerRIF: An interactive curation system for biomarker (H-J Dai)</li> <li>• 9:50 Supporting Document Triage with the SciKnowMine System in the Mouse Genome Informatics (MGI) Curation Process (G. Burns)</li> <li>• 10:00 BioQRator: a web-based interactive biomedical literature curating system (S. Kim/C. Arighi)</li> <li>• 10:10 RLIMS-P: Literature-based curation of protein phosphorylation information (M. Torii)</li> <li>• 10:20 Egas – Collaborative Biomedical Annotation as a Service ( S. Matos)</li> <li>• 10:30 tagtog: Interactive Human and Machine Annotation of Gene Mentions in PLOS Full-Text Articles (J.M. Cejuela)</li> <li>• 10:40 Customizable Curation Workflows in Argo (R. Navarro-Batista)</li> <li>• 10:50 Discussion:organizers &amp; all participating teams</li> </ul>
11:00 - 11:20	Break
11:20 - 12:50	<p><b>TRACK 5 IAT demo session</b></p> <ul style="list-style-type: none"> <li>• 11:20 CellFinder (M. Neves)</li> <li>• 11:30 Ontogene (F. Rinaldi)</li> <li>• 11:40 MarkerRIF (H-J Dai)</li> <li>• 11:50 SciKnowMine (G. Burns)</li> <li>• 12:00 BioQRator (S. Kim/C. Arighi)</li> <li>• 12:10 RLIMS-P (M. Torii)</li> <li>• 12:20 EGAS ( D. Campos)</li> <li>• 12:30 tagtog (J.M. Cejuela)</li> <li>• 12:40 Argo (R. Navarro-Batista)</li> </ul>
12:50 - 13:50	Lunch
13:50 - 14:50	<b>BioCreative - next steps</b>
14:50 - 15:00	<b>Workshop closing</b>



## TRACK 1 (BioC: Interoperability)

### Organizers:

- W. John Wilbur, National Center for Biotechnology Information (NCBI), NIH, USA
- Rezarta Islamaj Dogan, National Center for Biotechnology Information (NCBI), NIH, USA
- Donald C. Comeau, National Center for Biotechnology Information (NCBI), NIH, USA

# PyBioC: a python implementation of the BioC core

Hernani Marques, Fabio Rinaldi

## Motivation

BioC<sup>1</sup> (1) is a recently proposed framework which aims at providing a simple and yet powerful approach for the integration of text mining tools, based on a combination of an XML-based data interchange format, and the implementation of a library that allows memory-based handling of documents at all levels of processing. Implementations of the BioC framework have been provided in Java and C++.

OntoGene is a specialized text mining system for Named Entity Recognition (NER) and relationship extraction, capable of dealing with a variety of entities, such as chemicals, diseases, drugs, genes or proteins. The effectiveness of the system has been tested in several text mining evaluations, in which the OntoGene team has participated with success. Best results have been achieved in the detection of experimental methods (BioCreative 2006), in the detection of interactions between proteins (BioCreative 2009), in large-scale detection of biomedical entities for some entity categories (CALBC 2010). In the CTD triage task of BioCreative 2012 the OntoGene system provided best overall results.

The OntoGene system (3,4) is based on a relatively heterogeneous pipeline, composed of tools implemented in perl, xslt, python and prolog. The different modules of the pipeline exchange text annotations as I/O files in a common XML format, which has several similarities with the proposed BioC XML format. The original idea of the system was to allow memory-based sharing of annotations, but this was never implemented due to the heterogeneity of the components. Recently most of the modules (but not yet all of them) have been reimplemented in python, which would allow memory-based processing. We plan to switch gradually to a BioC compatible internal format and for this purpose we have decided to provide an independent implementation of the BioC core in python.

Additionally, as part of our participation in task 3, we have also implemented a RESTful (2) web service which allows submission of input documents in BioC format and delivers entity annotations in the same format.

---

<sup>1</sup> <http://www.ncbi.nlm.nih.gov/CBBresearch/Dogan/BioC/>

## **PyBioC**

In BioCreative IV Track-1 participants were asked to contribute to the BioC (BioCreative) community in the area of interoperability. The Ontogene team based in Zurich was confronted with the fact that no native BioCreative library for use with the Python programming language was available until now. To our knowledge no other team or initiative aimed at changing this, such that we took up this opportunity to create a Python implementation of the BioC library.

The PyBioC library recreates the functionality of the already available libraries in C++ or Java. However, we adhere to Python conventions where suitable, for example refraining from implementing getter or setter methods for internal variables of the classes provided in PyBioC.

Basically the library consists of a set of classes representing the minimalistic data model proposed by the BioC community. Two specific classes (BioCReader and BioCWriter) are available to read in data provided in (valid) XML format and to write from PyBioC objects to valid BioC XML format. Validity is ensured by following the BioC DTD publicly available.

The library is being developed as free software and is available on a public github repository (<https://github.com/2mh/PyBioC>), where example programs can be found, specifically to read in and write to BioC XML format or to tokenize and stem a given BioC XML input file using the Natural Language Toolkit (NLTK) library. As an example of an application, we also provide the integration of a standard word stemmer in PyBioC:

<https://github.com/2mh/PyBioC/blob/master/src/stemmer.py>

## **Conclusion**

PyBioC enables the biomedical text mining community to deal with BioC XML documents using a native implementation of the BioC library in the Python programming language. PyBioC is available as open-source under the Simplified BSD License. We welcome further contributions and additions to this work.

Our contribution to task 1 (a python implementation of the BioC core) is available at:

<https://github.com/2mh/PyBioC/tree/master/src/biocβ>

## **Acknowledgements**

The OntoGene group is partially supported by the Swiss National Science Foundation (grants 100014- 118396/1 and 105315- 130558/1 ).

## References

1. Donald C. Comeau, Rezarta Islamaj Doğan, Paolo Ciccarese, Kevin Bretonnel Cohen, Martin Krallinger, Florian Leitner, Zhiyong Lu, Yifang Peng, Fabio Rinaldi, Manabu Torii, Alfonso Valencia, Karin Verspoor, Thomas C. Wiegers, Cathy H. Wu, W. John Wilbur (2013). BioC: a minimalist approach to interoperability for biomedical text processing, *The Journal of Biological Databases and Curation* (2013), *bat064*, doi:10.1093/database/bat064, published online.
2. Richardson, Leonard; Ruby, Sam (2007), *RESTful Web Services*, O'Reilly, ISBN 978-0-596-52926-0
3. Fabio Rinaldi, Thomas Kappeler, Kaarel Kaljurand, Gerold Schneider, Manfred Klenner, Simon Clematide, Michael Hess, Jean-Marc von Allmen, Pierre Parisot, Martin Romacker, Therese Vachon (2008). OntoGene in BioCreative II. *Genome Biology*, 2008, 9:S13, PMC2559984
4. Fabio Rinaldi, Gerold Schneider, Kaarel Kaljurand, Simon Clematide, Thérèse Vachon, Martin Romacker, "OntoGene in BioCreative II.5," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 7(3), pp. 472-480, 2010. <http://doi.ieeecomputersociety.org/10.1109/TCBB.2010.50>

# Enhancing the Interoperability of iSimp by Using the BioC Format

Yifan Peng<sup>1,\*</sup>, Catalina O Tudor<sup>1,2</sup>, Manabu Torii<sup>1,2</sup>, Cathy H Wu<sup>1,2</sup>, K Vijay-Shanker<sup>1</sup>

<sup>1</sup>Department of Computer and Information Sciences, University of Delaware, Newark, DE,

<sup>2</sup>Center for Bioinformatics and Computational Biology, University of Delaware, Newark, DE

\*Corresponding author: Tel: 302 831 8496, E-mail: yfpeng@udel.edu

## Abstract

This paper reports the use of the BioC format in our sentence simplification system, iSimp, so that it could be seamlessly used in text mining pipelines. iSimp is designed to simplify complex sentences commonly found in the biomedical text, therefore bringing benefits to existing text mining applications that rely on the analysis of sentence structures. By adopting the BioC format, we aim to make iSimp readily integrable in various applications in this domain. To examine the utility of iSimp with BioC, we designed and implemented a rule-based relation extraction system that uses iSimp as a preprocessing module and BioC format for data exchange. Evaluation on the BioNLP-ST 2011 GE task training corpus showed that, with sentence simplification provided by iSimp, the F-value of the phosphorylation extraction increased 3%. The iSimp corpus previously used for the evaluation of simplification and the GE task corpus used in the current study have been converted into the BioC format and made publicly available<sup>1</sup>.

## Introduction

The syntactic complexity of the biomedical text often poses a major challenge in designing and applying Natural Language Processing (NLP) systems on scientific articles. One possible approach to address this issue and improve the performance of NLP systems (e.g., relation extraction systems) is to simplify the complexity of the sentences themselves prior to using them as input in the NLP systems. For this purpose, we had previously developed iSimp [1], a sentence simplification system. Used a preprocessing module that simplifies the input text, iSimp has a potential to enhance existing text mining applications in the biomedical domain. In order to make iSimp readily integrable in various applications, we have adopted the BioC format, a simple, yet robust, XML format to share text documents and annotations [2].

We report in this paper how BioC is used with iSimp. One of the contributions of this work is a BioC tag set for annotating iSimp outputs. Sentence simplification is a task that there is no

---

<sup>1</sup> <http://research.dbi.udel.edu/isimp/corpus/>

standard scheme for annotating simplification results. We define a BioC tag set to share and compare simplification annotation results from various simplifiers.

A second contribution of this work is a mechanism, using the BioC framework, to encode simplified sentences in the corpora. A factor that makes integration of iSimp with BioC format distinct, compared to many NLP tasks, is that the annotation can include sequences and words that are not from the original text. This is because iSimp produces new sentences together with annotation of the simplification constructs in the original text. Thus, the proposed mechanism allows simplified sentences to be included in a BioC annotation file and be treated as part of the original collection for further processing in the NLP pipeline.

A third contribution of this work is the iSimp corpus [1], which consists of 130 Medline abstracts and is annotated with six simplification constructs. We converted the corpus into BioC format and made it public available to be used for evaluation of different simplification systems. In order to show the wide applicability of iSimp, we examined its impact on event extraction. We designed and developed a simple rule-based relation extraction system. We showed that with sentence simplification provided by iSimp, the performance of the relation extraction system improves. We also present how iSimp can be utilized with BioC, by enabling both iSimp and the relation extraction system use BioC format. This makes the module integration seamlessly. For this, we report another contribution of this work, namely the conversion of the BioNLP-ST 2011 GE corpora into the BioC format and its public availability.

## Methods

In this section, we describe iSimp, the relation extraction, the corpus used in our evaluation, and how the BioC format is used to facilitate an easy I/O exchange between these components.

### iSimp

iSimp is designed to reduce the sentence syntactic complexity. To illustrate the usefulness of our sentence simplifier, iSimp, consider the following complex sentence:

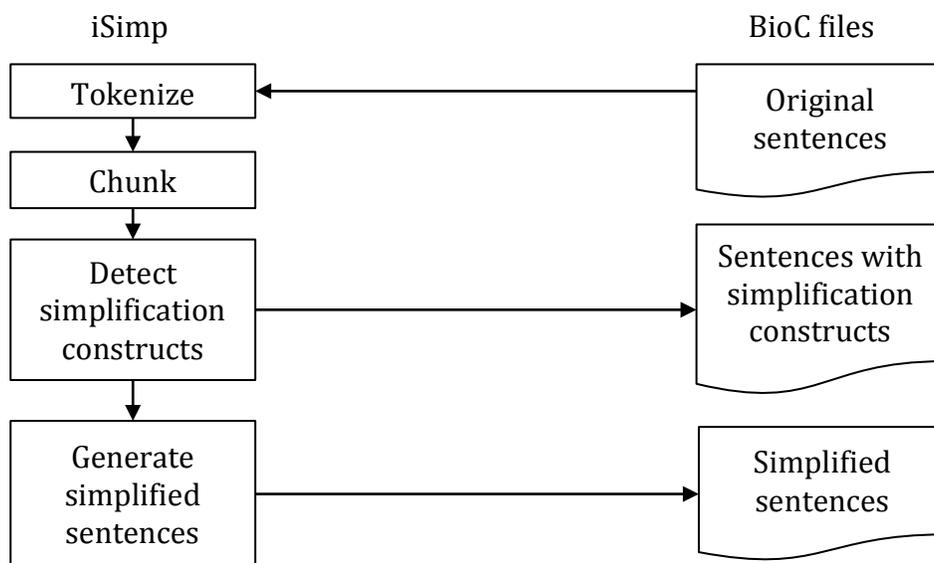
E1. A third genetic linkage to disease is alpha-synuclein, a protein that is heavily phosphorylated in Lewy bodies and Lewy neuritis, the pathological hallmarks of PD. (PMID-22342821)

As shown in this example, the major syntactic constructs that we considered for simplification are: coordination (e.g., “Lewy bodies and Lewy neuritis”), relative clause (e.g., “a protein that is heavily phosphorylated in ...”), and apposition (e.g., “alpha-synuclein, a protein that is ...” and “Lewy bodies and Lewy neuritis, the pathological hall marks of PD”). For a more detailed description of iSimp, as well as its challenges (attachment ambiguities, boundary detection, and nested constructs), we refer the reader to [1].

iSimp identifies the various types of simplifications ( coordinations, relative clauses, appositions, etc.) and breaks the complex sentence into multiple simple sentences. Here we only show two examples of simplifying (E1):

- E2. Alpha-synuclein is heavily phosphorylated in Lewy bodies.
- E3. Alpha-synuclein is heavily phosphorylated in Lewy neuritis.

We made iSimp available as an online tool<sup>2</sup>, and adopted the BioC format as its input/output format. Figure 1 shows the workflow of the system. iSimp first tokenizes the text and then it splits each sentence into a sequence of non-overlapping chunks. The detection of various simplification constructs is based on the chunks, and from these, iSimp then generates simplified sentences.



**Figure 1.** The workflow of iSimp

We see iSimp as a module to be used in the beginning of text mining applications. Developers can either use the webpage to submit input sentences in BioC format, or they can send POST requests to the service. The later technique will make iSimp easier to be integrated into other systems. Two types of output are provided: (1) sentences marked with simplification constructions, and (2) a list of simplified sentences, where each token is mapped back to the original text. It is often the case that new tokens will be added in simplified sentences to ensure their syntactical correctness. These new tokens will absolutely not be mapped to the original text.

### BioC Format in iSimp

BioC [2] is an XML format that ensures interoperability among documents and annotations, such as part-of-speech tags, name entities, or relations. Because sentence simplification requires a

<sup>2</sup> <http://research.bioinformatics.udel.edu/isimp/services.html>

unique schema for annotation, unlike most NLP tasks, we define a BioC tag set for annotating and sharing the simplification results. We use “BioCAnnotation” to mark the simplification construct components, e.g., conjuncts and conjunctions in coordinations. We use “BioCRelation” to specify how they are related. In this way, we are able to assign roles for each component and skip over symbols like comma.

Additionally, iSimp poses a challenge to the BioC format because it also generates new simplified sentences. Such challenges were not discussed in [2]. The BioC XML file generated by iSimp contains both original and simplified sentences. Original sentences' offsets are the same as in the original text. However, simplified sentences' offsets start with the next char after the last in the original document (last document's offset + last document's length). This new collection could then be treated as the input collection for further processing in the NLP pipeline.

In order to link text in simplified sentences to that in the original sentence, we provide “equivalence” relations, which can be helpful for information extraction tasks. For example, we link “alpha-synuclein” and “phosphorylated” in both (E2) and (E3) back to (E1). Thus, only one relation <alpha-synuclein, phosphorylated> will be extracted from (E1)-(E3). This technique makes iSimp different from previous sentence simplification systems such as BioSimplify [3].

### Relation extraction system

To examine the usefulness of iSimp, we designed and developed a rule-based relation extraction system. The first relation we focused on was the phosphorylation relation. We manually created a list of rules, where X is a protein or protein product. Some example rules are shown below:

1. phosphorylation of X
2. X phosphorylation
3. [noun phrase phosphorylated X]
4. phosphorylate (or, phosphorylates, phosphorylated, phosphorylating) X

These rules are able to match simple mentions of phosphorylation in text, however they will fail to match phosphorylation mentions in complex sentences, like the one shown below.

E4. However, the activated pAkt did not lead to [<sub>coordination</sub> **phosphorylation and inactivation**] of the downstream target GSK3 (PMC-2065877).

But iSimp is able to generate two simple sentences from (E4):

- E5. However, the activated pAkt did not lead to **phosphorylation** of the downstream target GSK3.  
E6. However, the activated pAkt did not lead to **inactivation** of the downstream target GSK3.

The first rule above can now apply on (E5) and extract <phosphorylation, GSK3>. Because the hand-crafted rules are very precise, the simplification step will only help improve the recall of the system, without hurting the precision.

We have converted the BioNLP-ST 2011 GE corpus to the BioC format for evaluation purposes. The training set, the development set, as well as the conversion script are now publicly available. The test set was not included in the release because it does not contain event annotations. Jimeno Yepes, et al. [5] discusses convention between the brat and BioC format.

## Results

For others to evaluate the performance of iSimp, we provide a corpus marked with simplification constructs, using the BioC format (<http://research.bioinformatics.udel.edu/isimp/corpus.html>). To examine the usability of iSimp in other systems, we tested the relation extraction system on the BioNLP-ST 2011 GE task training corpus. Results show that the Precision/Recall/F-value of the phosphorylation extraction before and after simplification are 97.32/78.38/86.83 versus 97.42/81.62/88.82, respectively. Therefore, with the help of iSimp, the recall of the relation extraction system improved by 3%, while the precision stayed the same. In the ongoing work, we have observed similar improvement in the recall for other relation extraction tasks.

## Conclusion

In order to participate in the BioCreative IV track 1, we adapted our previously developed system, iSimp (a sentence simplification system), to read and write BioC format files. We converted a previously annotated corpus to the BioC format to show the performance of iSimp. To emphasize the wide applicability of iSimp, we examined its impact on event extraction. We released the simplification corpus, the BioNLP corpus, and the conversion script, for others to easily judge the results and use them in comparing and designing other simplifiers.

## Funding

This work was supported by the NLM of NIH [G08LM010720] and NSF [DBI-1062520].

## References

1. Peng, Y., Tudor, C.O., Torii, M., Wu, C.H. and Vijay-Shanker, K. (2012) iSimp: A sentence simplification system for biomedical text. *In Proceedings of the 2012 IEEE International Conference on Bioinformatics and Biomedicine (BIBM 2012)*, 211-216.
2. Comeau, D.C., Dogan, R.I., Ciccarese, P., Cohen, K.B., Krallinger, M., Leitner, F., Lu, Z., Peng, Y., Rinaldi, F., Torii, M., Valencia, V., Verspoor, K., Wiegers, T.C., Wu, C.H., and Wilbur, W.J. (2013) BioC: A minimalist approach to interoperability for biomedical text processing. *Database: The Journal of Biological Databases and Curation*.
3. Jonnalagadda, S. and Gonzalez, G. (2010) BioSimplify: An open source sentence simplification engine to improve recall in automatic biomedical information extraction. *AMIA Annual Symposium Proceedings*.
4. Kim, J.D., Nguyen, N., Wang, Y., Tsujii, J., Takagi, T. and Yonezawa, A. (2012) The Genia event and protein coreference tasks of the BioNLP shared task 2011. *BMC Bioinformatics*, 13 (Suppl 11):S1.
5. Jimeno Yepes, A., Neves, M. and Verspoor, K. (2013) Brat2BioC: conversion tool between brat and BioC. Submitted to the BioCreative IV workshop.

# Improving Interoperability of Text Mining Tools with BioC

Ritu Khare, Chih-Hsuan Wei, Yuqing Mao, Robert Leaman, Zhiyong Lu\*

National Center for Biotechnology Information, 8600 Rockville Pike, Bethesda, Maryland, USA

\*Corresponding Author: Tel: 301-594-7089, E-mail: zhiyong.lu@nih.gov

## Abstract

The lack of interoperability among text mining tools is a major bottleneck in creating more complex applications. Despite the availability of numerous methods and techniques for various text mining tasks, combining different tools requires substantial efforts and time. In response, BioC offers a minimalistic approach to tool interoperability by stipulating minimal changes to existing tools and applications. In this study, we introduce several state-of-the-art text mining tools (for recognizing and annotating genes, diseases, mutations, species, and chemicals in biomedical text) developed at the National Center for Biotechnology Information (NCBI), and modify these tools to make them BioC compatible. We find that only minimal changes were required in order to build the BioC versions of our tools via using the BioC family of format and functions. Through this work, we improved the interoperability of our tools, and anticipate serving a wider community for building more sophisticated applications. Our toolkit was created through participating in the BioCreative IV Interoperability Track and is publicly available at <http://www.ncbi.nlm.nih.gov/CBBresearch/Lu/Demo/tmTools>.

## Introduction

There is an increasing demand of text mining tools in the biomedical and life sciences domain. Many recent BioNLP challenge tasks (1-5) are focused on extracting structured information from scientific articles and clinical notes. Research groups around the world are developing a variety of standalone text mining tools. Typically, a tool is developed using a certain preferred data representation, programming conventions as determined by the individual research group. In order to build complex text mining applications or pipelines, it is often required to combine multiple tools, possibly designed by different groups. The current practice of independent tool development poses a hindrance to tool interoperability and integration. In order to use a new tool or a new dataset, text mining researchers spend a substantial amount of time developing algorithms for processing the new data format. This heterogeneity in data representations slows down the development of powerful applications and thereby leads to inefficiencies in research and innovation.

There have been quite a few efforts to promote interoperability among text analytics tools. Unstructured information management architecture (UIMA)(6-8) and General Architecture for

Text Engineering (GATE) (9) are two notable proposals that prescribe using a predefined framework to develop text mining applications to achieve interoperability among independently developed tools. Development of UIMA- or GATE-compliant applications requires the entire tool to be (re-)written into framework specific constructs. The steep learning curve associated with these frameworks keeps them from being broadly accepted as a development and data sharing standard (10). Motivated by this, a recent effort in this direction, BioC (11), is based on a minimalist approach in that it offers interoperability by stipulating minimal changes in existing applications or datasets. The goals of BioC are simplicity, reusability, interoperability and wide use. In a nutshell, BioC is a family of XML formats that define how to present text documents and annotations. BioC also provides tools to read and write documents in the BioC format in two widely used programming languages.

In this paper, we present our efforts on using BioC to re-package the suite of text mining software and web-based tools (12-18) developed at the biomedical text mining group at the National Center for Biotechnology Information (NCBI). Specifically, we wrap five stand-alone biomedical named entity recognition (NER) tools, one web-based annotation tool, and one annotated text corpus, into BioC. All tools are aimed toward accelerating the biomedical discovery and manual curation of biological databases, and by making them BioC compatible, we expect them to serve a wider community.

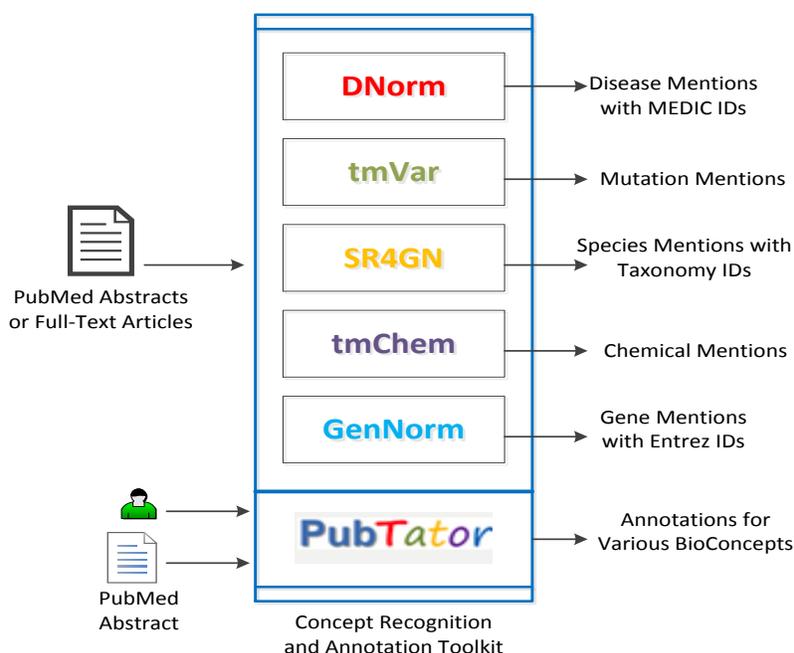
## **Method**

In this section, we first introduce the NCBI suite of tools that comprises six tools for concept recognition and annotation, and an annotated text corpus for Gene Ontology concept recognition. Then, we describe the key steps and challenges in creating a BioC compatible version of the tools and the text corpus.

### **Our Concept Recognition and Annotation Toolkit**

At NCBI, we have developed several NER tools for automatically recognizing key biomedical concepts such as chemicals, diseases, genes, mutations, and species, from scientific publications. Each tool accepts a PubMed or PMC full-text article as an input and identifies the biomedical entities at either mention-level, or at both mention and concept level. Figure 1 provides a summary of our concept recognition and annotation toolkit.

- *DNorm*(1,15) is an open-source software tool to identify and normalize disease mentions from biomedical texts. *DNorm* is based on pair-wise learning to rank and is the first technique to use machine learning for disease normalization. This tool was developed in Java.



**Figure 1.** Visual Summary of NCBI Concept Recognition and Annotation Toolkit

- *tmVar*(14) is a machine learning system for mutation recognition to assist biomedical curation. It is based on conditional random fields and identifies many types of mutations and sequence variants in protein, gene, DNA, and RNA levels for biomedical curation. This tool was developed in Perl and uses the CRF++ module developed in C++.
- *SR4GN*(12) is a species recognition tool optimized for the gene normalization task. It is a rule-based system that identifies species from full-texts and pairs them with corresponding gene or protein mentions. This tool was developed in Perl.
- *tmChem*(17) is a machine learning based NER system for chemicals. The system is designed to identify a wide variety of chemical mentions in literature, including identifiers, brand and trade names and also systematic formats. The system uses conditional random fields with a rich feature set and rule-based post processing modules for resolving local abbreviations and improving consistency. This tool was developed in Java.
- *GenNorm*(13) is a rule-based tool to for gene recognition and performs gene name recognition, species assignment and species-specific gene normalization. *GenNorm* addresses the challenging issues of orthologous gene ambiguity and intra-species gene ambiguity. This tool was developed in Perl.

Based on the above NER tools, we also developed a web-based annotation tool called *PubTator* (16,19,20) for assisting manual curation. *PubTator* is in sync with PubMed and supports annotation of biomedical entities and their relationships in PubMed articles.

### **The BC4GO corpus**

More recently, we developed the *BC4GO* corpus (18) (not shown in the figure), a corpus of 200 full-text articles along with their gene ontology (GO) annotations describing genes and gene product attributes across species and databases. As annotations, the corpus presents the evidence sentences along with the gene/protein entities, GO terms, and GO evidence codes. The corpus was developed with eight expert biocurators using a web-based annotation tool. This is the official corpus for the BioCreative IV Track-4 GO Task (21), which tackles the challenge of automatic GO annotation through literature analysis.

### **Building BioC Compatible Tools**

The BioC family of XML formats and functions comprises the following three items:

- (i) The XML Document Type Definition (DTD) that defines how to present text document and annotations in higher-level semantics to share common information. It allows many different annotations to be represented, including sentences, tokens and named entities. The general BioC format recommends keeping the text of the article and the corresponding annotations in separate files, namely BioC article file and BioC annotation file.
- (ii) A key file to describe the lower level semantics of the elements in the BioC annotation file. The key file describes how data should be interpreted in the BioC annotation file, and needs to be created for a specific type of data
- (iii) C++ and Java libraries that include functions and classes to read and write documents in BioC format and to hold the documents in memory.

To comply our tools with BioC, we modified the input and output formats of the tools, i.e., by adding BioC as a new option, and translated the articles and the annotations into BioC article files and BioC annotation files, respectively.

### **Concept Recognition Tools**

The main challenge faced when converting these various concept recognition tools to BioC was to define an appropriate key file. Since the semantics of all these tools are similar to *PubTator* in terms of the type of data, we used the same key file, **PubTator.key**. The same key file is used for interpreting the input full-text articles/abstracts, and the output articles/abstracts with annotations.

The mutation recognition tool, *tmVar*, originally accepts the PubTator format, free text, and the PMC XML format. The output format is the PubTator format. For *GenNorm* and *SR4GN*, the input formats are free text, PMC XML format, and the GenNorm format<sup>1</sup>, and the output format is the GenNorm format. To make these tools compatible with BioC, we added the BioC format

---

<sup>1</sup> <http://www.ncbi.nlm.nih.gov/CBBresearch/Lu/Demo/Summary/Format.html#GenNorm>

as a new option for input/ output. The main difference between the custom-defined format and the BioC format is the offset calculation. The custom-defined format calculates the offsets for separate sentences, and to translate to the BioC format, we had to re-calculate the global offset for the each mention. Accordingly, we added the new elements, mention, and paragraph, in the key file.

The previous output format for *tmChem* was the BioCreative IV CHEMDNER format, which is essentially a delimited format for representing one NER mention on each line. *DNorm* is a relatively new tool and did not previously have a default output format. Since both tools are built on top of BANNER (22), input compatibility with BioC only required writing a single new dataset loading class in BANNER to read BioC. Modifying the output required modifying the class containing the main method to output the BioC format.

### ***PubTator***

The original input/output format for *PubTator* is a pre-defined format that we refer to as the PubTator format<sup>2</sup>. To make *PubTator* BioC compatible, we added a new format option giving users the option to input and output in the BioC format. For our purposes, we also slightly modified the original BioC format. The original BioC recommends keeping only the annotations, and not the passages, in the BioC annotation file. However, this would require users to upload two files when importing annotations to *PubTator*. Hence, in our version of BioC compatible *PubTator*, the annotations are appended after the article passages in the BioC annotation file. The other concepts recognition tools and corpus still follow the original BioC format. We defined the **PubTator.key** file that describes specific attributes such as bioconcept, identifier, offset, and mentions.

### ***BC4GO Corpus***

First, the 200 full-text articles of the *BC4GO* corpus, originally in the PMC XML data model format, were converted to the BioC format. Then, we extracted annotated sentences from downloaded HTML files from the tool and identified their offsets. Finally, for each article we created a corresponding BioC annotation file for the associated GO annotations. For the gene entity, we provide both the gene mention as appeared in text and its corresponding NCBI Gene identifier. In the BioC released corpus, each article is named by its PubMed identifier, e.g. “20130316.xml.” The annotation file associated with the article file shares the same PMID in the file name, e.g. “annotation\_20130316.xml.” The annotation file includes all annotations of the article; each annotation has a unique ID and is defined by four distinct elements: gene, go-term, go-evidence, and type. We define separate key files to describe the full-text articles and the annotation files with GO annotations, namely **pmc\_go.key** and **go\_annotation.key**, respectively. There were certain challenges in creating the BioC compatible version of the *BC4GO* corpus.

---

<sup>2</sup> <http://www.ncbi.nlm.nih.gov/CBBresearch/Lu/Demo/PubTator/import.example.html>

```

<collection>
  <source>Example</source>
  <date>1999-Jan-1</date>
  <key>PubTator.key</key>
  <document>
    <id>20085714</id>
    <passage>
      <infony key="type">title</infony>
      <offset>0</offset>
      <text>Autosomal-dominant striatal degeneration is caused by a mutation in the
        phosphodiesterase 8B gene.</text>
    </passage>
    <passage>
      <infony key="type">abstract</infony>
      <offset>98</offset>
      <text>Autosomal-dominant striatal degeneration (ADSD) is an autosomal-dominant movement
        disorder affecting the striatal part of the basal ganglia. ADSD is characterized by
        bradykinesia, dysarthria, and muscle rigidity. These symptoms resemble idiopathic
        Parkinson disease, but tremor is not present. Using genetic linkage analysis, we
        have mapped the causative genetic defect to a 3.25 megabase candidate region on
        chromosome 5q13.3-q14.1. A maximum LOD score of 4.1 (Theta = 0) was obtained at
        marker D5S1962. Here we show that ADSD is caused by a complex frameshift mutation
        (c.94G>C+c.95delT) in the phosphodiesterase 8B (PDE8B) gene, which results in a loss
        of enzymatic phosphodiesterase activity. We found that PDE8B is highly expressed in
        the brain, especially in the putamen, which is affected by ADSD. PDE8B degrades
        cyclic AMP, a second messenger implied in dopamine signaling. Dopamine is one of the
        main neurotransmitters involved in movement control and is deficient in Parkinson
        disease. We believe that the functional analysis of PDE8B will help to further
        elucidate the pathomechanism of ADSD as well as contribute to a better understanding
        of movement disorders.</text>
    </passage>
  </document>
</collection>

```

**Figure 2.** A snippet of the BioC article file for PMID 20085714

The first challenge was in creating the BioC annotation file using the user annotations downloaded from the web-based annotation tool. We observed encoding discrepancies in the article file and the downloaded file. The original file in PMC XML format is encoded in ASCII, which is also the encoding convention for the BioC format. However, the annotation results downloaded from the Web were encoded using Unicode. For example, the term “neurexin-1 $\alpha$ ” (see PMID:22262843 in corpus) would read “neurexin-1alpha” in ASCII but “neurexin-II+” in Unicode. In order to maintain consistency between the BioC article and annotation files, we translated the Unicode characters back to ASCII using a neighbor matching method as described in (18).

Another challenge was presenting those evidence sentences that contain multiple discontinuous sentences, possibly from different passages in the article (see the evidence sentence for GO:1990124 in PMID 18695045 in corpus). We addressed this challenge by linking these disjoint evidence sentences using the same annotation ID for recognition, i.e., they are treated as one whole evidence sentence for a GO term.

```

<annotation>
  <infun key="type">Mutation</infun>
  <offset>679</offset>
  <length>8</length>
  <text>c.95delT</text>
  <id>cIDELI95IT</id>
</annotation>
<annotation>
  <infun key="type">Gene</infun>
  <offset>696</offset>
  <length>20</length>
  <text>phosphodiesterase 8B</text>
  <id>8622</id>
</annotation>
<annotation>
  <infun key="type">Gene</infun>
  <offset>718</offset>
  <length>5</length>
  <text>PDE8B</text>
  <id>8622</id>
</annotation>
<annotation>
  <infun key="type">Gene</infun>
  <offset>810</offset>
  <length>5</length>
  <text>PDE8B</text>
  <id>8622</id>
</annotation>
<annotation>
  <infun key="type">Disease</infun>
  <offset>898</offset>
  <length>4</length>
  <text>ASD</text>
  <id>609161</id>
</annotation>
<annotation>
  <infun key="type">Gene</infun>
  <offset>904</offset>
  <length>5</length>
  <text>PDE8B</text>
  <id>8622</id>
</annotation>
<annotation>
  <infun key="type">Chemical</infun>
  <offset>919</offset>
  <length>10</length>
  <text>cyclic AMP</text>
</annotation>

```

**Figure 3.** A snippet from the BioC annotation file for PMID 20085714 (integrated result of applying our five concept recognition tools on the abstract). The offset element is the global offset.

One limitations of the corpus released in BioC is that the BioC annotation file of an article would not contain an evidence sentence that is located in the “Acknowledgement” section of the article (see PMID 18695045) because this section is not provided in the original PMC XML file for the article. Also, in some cases, such as footnotes, incomplete sentences were created due to the additional space characters in the original PMC XML files. Such cases were manually processed to create a consistent BioC annotation files.

## Results

The new BioC versions of all tools and the common **PubTatory.key** file are made publicly available. The key file is also shown in Figure 5 in the Appendix section.

To describe the outputs of our concept recognition tools, we use a PubMed abstract (PMID 20085714) that contains mentions of multiple biomedical entities, including genes, mutations, chemicals, and diseases, as a running example. A snippet of the BioC article file for this example is shown in Figure 2, and the integrated results from all the tools are displayed in Figure 3 showing a snippet of the BioC annotation file.

The BioC version of the *BC4GO* corpus, with 200 BioC article files and 200 BioC annotation files, can be downloaded at the BioCreative IV Track 4 task’s official webpage, <http://www.biocreative.org/tasks/biocreative-iv/track-4-GO/>. The key files, **pmc\_go.key** and **go\_annotation.key** are submitted as part of the BioCreative IV Track 1 submission. These key files are also shown as Figures 6 and 7 in the Appendix section. A snippet of the BioC annotation file corresponding to the PubMed article with PMID 23840682 is shown in Figure 4.

```
<collection>
  <source>GO_Annotation</source>
  <date>20130316</date>
  <key>go_annotation.key</key>
  <document>
    <id>23840682</id>
    <passage>
      <infor key="type">abstract</infor>
      <offset>89</offset>
      <annotation id="23840682_1">
        <infor key="gene">emb16(100170235)</infor>
        <infor key="go-term">embryo development|GO:0009790</infor>
        <infor key="goevidence">IMP</infor>
        <infor key="type">GOA</infor>
        <location offset="415" length="114"/>
        <text>The emb16 mutation arrests embryogenesis at transition stage and allows the
          endosperm to develop largely normally.</text>
      </annotation>
    </passage>
  </document>
</collection>
```

**Figure 4.** A Snippet from the file **annotation\_23840682.xml** from the *BC4GO* corpus

PubTator.key: A BioC format for PubTator and all equipped tools (i.e., tmChem, DNorm, tmVar, SR4GN or GenNorm).

The goal of this collection is to provide easy access to the text and its bio-concept annotation of PMC articles. All of the text in an article is easily accessible. Some of the other information in an article is also available.

collection: a group of PubMed documents split into title, abstract and other passages

source: PubMed or PubMed Central

date: Date document downloaded from PubTator

document: Title, abstract and other passages from a PubMed or PMC reference

id: PubMed id

passage: Title, abstract and other passages

infor["type"]: "title", "abstract" and other passages

offset: Title has an offset of zero, while the other passages (e.g., abstract) are assumed to begin after the previous passages and one space

annotation: One bio-concept of the passage as determined by the tmChem, DNorm, tmVar, SR4GN or GenNorm

infor["type"]: "Gene", "Species", "Disease", "Chemical" or "Mutation"

id: The bio-concept identifiers which are detected by DNorm,tmVar, SR4GN and GenNorm

offset: A document offset to where the bio-concept begins in the passage. The global offset within the document

length: The length of the bio-concept in the passage

text: Mention of the bio-concept

**Figure 5.** The **PubTator.key** file

## Discussion and Conclusions

The goal of this study was to improve the interoperability of our NER tools using the recently developed BioC Family of XML formats and classes. The NCBI suite of tools consists of several competition winning, high-performing tools for concept recognition and annotation. For example, *GenNorm* obtained the highest performance in the BioCreative III Gene Normalization task (23), and *DNorm* achieved the best results the 2013 ShARe/CLEF shared task for

normalizing disease names in clinical notes (1). Also, the *tmVar* tool for mutation recognition delivers over 90% F-measure on multiple benchmarking test sets; and the PubMed-like, color-coded interface of *PubTator* makes it a highly usable annotation tool for human biocurators. In addition to accelerating knowledge discovery and assisting manual curation, the NCBI text mining toolkit is capable of solving other important and challenging problems in the biomedical domain. For instance, text mining mutation information is very critical for the analysis and interpretation of sequence variations in complex diseases in the post-genomic era. Disease recognition is important for many lines of inquiry, including etiology (e.g. gene-disease relationships) and clinical aspects (e.g. diagnosis, prevention, and treatment). Gene and species recognition could be useful for protein-protein interaction extraction.

`pmc_go.key`: A BioC format for PubMed Central (PMC) articles.

The goal of this collection is to provide easy access to the full-text of PMC articles.

collection: PMC articles articles selected for the GO annotation track of BioCreative IV

source: PMC

date: yyyyymmdd. Date articles downloaded from PMC.

document: PMC article

id: PubMed id

passage: Title, abstract and other passages

`infol["type"]`: "title", "abstract" and other passages

`offset`: Title has an offset of zero, while the other passages (e.g., abstract) are assumed to begin after the previous passages and one space

`text`: The ASCII text of the passage.

**Figure 6.** The `pmc_go.key` file

go\_annotation.key: A BioC format for PubMed Central (PMC) article annotations.

The goal of this collection is to provide easy access to the text of PMC article annotations. All of the text in an article is easily accessible. Some of the other information in an article is also available.

collection: Annotations of PMC articles articles selected for the GO annotation track of BioCreative IV

source: PMC and GO annotations made by professional GO curators

date: yyyyymmdd. Date articles downloaded from PMC.

document: PMC article

id: PubMed id

passage: Title, abstract and other passages

infor["type"]: "title", "abstract" and other passages

offset: Title has an offset of zero, while the other passages (e.g., abstract) are assumed to begin after the previous passages and one space

annotation: The evidence sentence of the passage as determined by the curator

infor["type"]: "gene", "go-term", "goevidence" and "type" of the annotation (typically "GOA").

offset: A document offset to where the evidence sentence begins in the passage. The global offset within the document

length: Length of the evidence sentence

text: ASCII text of the evidence sentence

**Figure 7.** The `go_annotation.key` file

Our experience shows that only minimal changes were required to re-package the NCBI suite of text mining tools with BioC. Also, reading and writing to BioC format was fairly straightforward as the functions and classes are already provided by the BioC authors in two widely used programming languages. For each tool, the primary developers modified their respective tools, and confirmed the simplicity and learnability of the BioC format. The primary challenge was to create the key files for the tools. However, it was a one-time effort since all the six concept

recognition and annotation tools can use a common key file for defining their BioC annotation files. The released **PubTator.key** file could also evolve as a standard key file for concept recognition and annotation tasks as recommended in (24). All our tools are freely available and ready to be re-used by a wider community of researchers in text mining, bioinformatics, and biocuration communities.

Through this study, we promote the interoperability of our tools, not only with each other, but also with the tools and datasets developed by several other groups worldwide. The tools, although developed in different programming languages such as Java, Perl, and C++, are now capable of sharing their inputs/outputs with each other, without any additional programming efforts. Our tools in BioC can interact with other state-of-the-art tools to build much more powerful applications. For example, a modular text mining pipeline of various BioC compatible tools for NER, normalization, annotation, and relationship extraction, could be developed to build sophisticated systems, e.g., an integrative disease-centered system connecting the biological and clinical aspects, providing information from causes (gene-mutation-disease relationship) to treatment (drug-disease relationships) of diseases by mining/annotating unstructured (biomedical literature, clinical notes, etc.) and structured resources (datasets released by organizations and research groups). In the future, we anticipate much broader usage of these tools as further efforts are invested in publicizing BioC.

## Acknowledgements

We would like to thank Don Comeau, Rezarta Dogan and John Wilbur for their discussion and help with the BioC tools and in particular, their help on preparing the PMC articles in BioC XML format for the BioCreative IV GO task. This research was supported by the Intramural Research Program of the NIH, National Library of Medicine.

## References

1. Leaman, R., Khare, R., Lu, Z. (2013) NCBI at 2013 ShARe/CLEF eHealth Shared Task: Disorder Normalization in Clinical Notes with DNorm. *Conference and Labs of the Evaluation Forum 2013 Working Notes*.
2. Lu, Z., Kao, H.Y., Wei, C.H., *et al.* (2011) The gene normalization task in BioCreative III. *BMC bioinformatics*, **12 Suppl 8**, S2.
3. Morgan, A.A., Lu, Z., Wang, X., *et al.* (2008) Overview of BioCreative II gene normalization. *Genome biology*, **9 Suppl 2**, S3.
4. Krallinger, M., Vazquez, M., Leitner, F., *et al.* (2011) The Protein-Protein Interaction tasks of BioCreative III: classification/ranking of articles and linking bio-ontology concepts to full text. *BMC bioinformatics*, **12 Suppl 8**, S3.
5. Mork, J.G., Bodenreider, O., Demner-Fushman, D., *et al.* (2010) Extracting Rx information from clinical narrative. *Journal of the American Medical Informatics Association : JAMIA*, **17**, 536-539.

6. Ferrucci, D., Lally, A. (2004) UIMA: An Architectural Approach to Unstructured Information Processing in the Corporate Research Environment. *Natural Language Engineering*, **10**, 327-348.
7. Ferrucci, D., Lally, A., Gruhl, D., *et al.* (2006) Towards an Interoperability Standard for Text and Multi-Modal Analytics. IBM Research.
8. Kano, Y., Baumgartner, W.A., Jr., McCrohon, L., *et al.* (2009) U-Compare: share and compare text mining tools with UIMA. *Bioinformatics*, **25**, 1997-1998.
9. GATE : General Architecture for Text Engineering. The University of Sheffield.
10. Stubbs, A. (2011) MAE and MAI: lightweight annotation and adjudication tools. *Proceedings of the 5th Linguistic Annotation Workshop*, pp. 129-133.
11. Comeau, D.C., Doğan, R.I., Ciccarese, P., *et al.* (2013) BioC: A Minimalist Approach to Interoperability for Biomedical Text Processing. *Database: The Journal of Biological Databases and Curation*.
12. Wei, C.H., Kao, H.Y., Lu, Z. (2012) SR4GN: a species recognition software tool for gene normalization. *PLoS one*, **7**, e38460.
13. Wei, C.H., Kao, H.Y. (2011) Cross-species gene normalization by species inference. *BMC bioinformatics*, **12 Suppl 8**, S5.
14. Wei, C.H., Harris, B.R., Kao, H.Y., *et al.* (2013) tmVar: a text mining approach for extracting sequence variants in biomedical literature. *Bioinformatics*, **29**, 1433-1439.
15. Leaman, R., Islamaj Dogan, R., Lu, Z. (2013) DNorm: Disease Name Normalization with Pairwise Learning to Rank. *Bioinformatics*.
16. Wei, C.H., Kao, H.Y., Lu, Z. (2013) PubTator: a web-based text mining tool for assisting biocuration. *Nucleic acids research*, **41**, W518-522.
17. Leaman, R., Wei, C.-H., Lu, Z. (2013) NCBI at the BioCreative IV CHEMDNER Task: Recognizing chemical names in PubMed articles using tmChem. *Proceedings of BioCreative IV*.
18. Auken, K.V., Schaeffer, M.L., McQuilton, P., *et al.* (2013) Corpus Construction for the BioCreative IV GO Task. *Proceedings of BioCreative IV*.
19. Wei, C.H., Harris, B.R., Li, D., *et al.* (2012) Accelerating literature curation with text-mining tools: a case study of using PubTator to curate genes in PubMed abstracts. *Database : the journal of biological databases and curation*, **2012**, bas041.
20. Wei, C.-H., Kao, H.-Y., Lu, Z. (2012) PubTator: A PubMed-like interactive curation system for document triage and literature curation. *Proceedings of BioCreative 2012 workshop*, Washington DC, USA, pp. 145-150.
21. Mao, Y., Auken, K.V., Li, D., *et al.* (2013) The Gene Ontology Task at BioCreative IV. *Proceedings of the BioCreative IV Workshop*, Bethesda, MD.
22. Leaman, R., Gonzalez, G. (2008) BANNER: an executable survey of advances in biomedical named entity recognition. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, 652-663.
23. Wei, C.-H., Kao, H.-Y. (2010) Inference network method on cross species gene normalization in full-text articles. *Proceeding of BioCreative III Workshop*, Bethesda, Maryland, pp. 73-81.
24. Arighi, C.N., Carterette, B., Cohen, K.B., *et al.* (2013) An overview of the BioCreative 2012 Workshop Track III: interactive text mining task. *Database : the journal of biological databases and curation*, **2013**, bas056.

# Finding Abbreviations in Biomedical Literature: Three BioC-Compatible Modules and Three BioC-formatted Corpora

Rezarta Islamaj Doğan, Donald C. Comeau, Lana Yeganova and W. John Wilbur

National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, 8600 Rockville Pike, Bethesda, MD 20894, MD, USA

\*Corresponding author: Tel: 301 435 5926, E-mail: wilbur@ncbi.nlm.nih.gov

## Abstract

This paper reports the use of BioC to address a common challenge in processing biomedical text information—that of frequent biomedical entity name abbreviation. We selected three different abbreviation definition identification modules, and used the publically available BioC code to convert these independent modules into BioC-compatible components that interact seamlessly with BioC-formatted data, and other BioC-compatible modules. In addition, we consider three manually-annotated corpora of abbreviations in biomedical text: the Ab3P corpus of 1250 PubMed abstracts, the BIOADI corpus of 1201 PubMed abstracts and the old MEDSTRACT corpus of 199 PubMed citations. Annotations in these corpora have been re-evaluated by four annotators and their consistency and quality levels have been improved. We convert them to BioC-format and describe the representation of their annotations. These corpora are used to test the three abbreviation finding algorithms. The BioC-compatible modules, when compared to their original form, have no difference in their efficiency, running time, or any other comparable aspects, so they can be conveniently used as a common pre-processing step for larger multi-layered text-mining endeavors.

Code and data are available for download at the BioC site: <http://bioc.sourceforge.net>.

## Introduction

The BioCreative<sup>1</sup> challenge evaluations since their inception in 2003 have been a community-wide effort for evaluating text mining information extraction systems applied to the biomedical domain. Given the emphasis on promoting scientific progress, BioCreative meetings have consistently sought to make available both suitable information extraction systems that handle life science literature and suitable “gold standard” data for training and testing these systems (1-4). The BioCreative IV Interoperability track<sup>2</sup> follows the guidelines established in previous BioCreative meetings, and specifically addresses the goal of interoperability—as a major barrier for wide-scale adoption of the developed text mining tools. As a solution, BioC (5) is proposed

---

<sup>1</sup> <http://www.biocreative.org/>

<sup>2</sup> <http://www.biocreative.org/tasks/biocreative-iv/track-1-interoperability/>

as an interchange format for tools for biomedical natural language processing. BioC is a simple XML format, specified by DTD, to share text documents and annotations. The BioC annotation approach allows many different annotations to be represented, including sentences, tokens, parts of speech, and named entities such as genes or diseases.

In this paper we present the contributions of our team to the BioC repository in the form of BioC-compliant modules that address the abbreviation definition detection task in biomedical text. These modules can be seamlessly coupled with other BioC code and used with any BioC-formatted corpora. We also present BioC-formatted corpora to test the abbreviation definition detection task, which can further be used with any other BioC-compliant tool for many biomedical natural language processing tasks.

### **Abbreviation detection in biomedical domain**

The past twenty years have only seen an increase in the interest for automatic extraction of biological information from scientific text, and particularly from MEDLINE abstracts. One characteristic of these documents is the frequent use of abbreviated terms. Abbreviated terms appear not only in the scientific text, but they are also frequent in user queries requesting the retrieval of those documents. Two related specific issues are: 1. The high rate at which new abbreviations are introduced in biomedical texts, and 2. The ambiguity of those abbreviations. Existing databases, ontologies, and dictionaries must be continually updated with new abbreviations and their definitions. In order to help resolve this problem, several techniques have been introduced to automatically extract abbreviations and their definitions from MEDLINE abstracts (6-9).

Abbreviation identification is the task of processing text to extract explicit occurrences of abbreviation-definition pairs. The task requires both the identification of sentences that contain *<short-form, long-form>* candidate pairs from text, and identification of exact long-form and short-form boundaries. An important clue that is shared by abbreviation detection methods is the presence of parenthetical text and the assumption that parenthetical text signals the presence of an abbreviation definition. Two cases are distinguished:

- a) long-form ‘(‘short-form‘)’, and
- b) short-form ‘(‘long-form‘)’, with the first alternative being observed much more frequently in practice.

An abbreviation—a short-form—is a shorter term that represents a longer word or phrase, which often refers to an important biomedical entity. The definition—the long-form—is searched for in the same sentence as the short-form, often between the beginning of the sentence and the open parenthesis, for case a).

## Abbreviation definition finding algorithms

### 1. The Schwartz and Hearst algorithm

The Schwartz and Hearst algorithm (7) decides about *<short-form, long-form>* candidates using this rule: If the expression within parentheses contains more than two words then case b) is assumed, otherwise case a) is assumed. A short-form is verified to contain the right number of characters, contains at least one letter character and starts with an alphanumeric character. A long-form candidate is extracted from the string so that it contains at the most  $\min(|SF|+5, |SF|*2)$  words, where  $|SF|$  is the number of characters in the short-form. The long-form is always longer in size than the short-form. Next, starting from the right end of both strings, the algorithm traverses both strings right to left, matching the characters in the short-form to find the shortest long-form string. The matching of characters has to be sequential in order for both strings. With the exception of the first short-form character that has to match a character at the beginning of a word in the long-form candidate, the rest of the characters can match anywhere in the long-form string, as long as they are in order. The algorithm is very simple, very fast, and the results are very good<sup>3</sup>.

### 2. The Ab3P Algorithm

The Ab3P algorithm, developed by Sohn et al., (6) is another pattern-matching approach to abbreviation definition detection. This algorithm defines 17 pattern-matching rules which the authors called strategies. Depending on the matching strategy and the length of the short form, they estimate an accuracy measure called pseudo-precision. Pseudo-precision provides a computed reliability estimate for an identified *<short-form, long-form>* pair, without any human judgment. This algorithm is also very fast and provides high precision results.

### 3. The NatLab algorithm

Rule-based methods, such as the Schwartz and Hearst algorithm and Ab3P, are successful at identifying abbreviation definition pairs with high precision. However, such approaches are unable to identify non-typical pairs, such as three dimensional (3-D), or out-of-order matches, such as melting temperature (T(m)). Machine learning methods have the potential of recognizing such non-trivial or irregular pairs and improving the recall, if enough training data is provided. NatLab (Natural Learning for Abbreviations), developed by Yeganova et al., (8) is an example of such algorithm.

NatLab is a supervised learning approach whose features, inspired by the basic rules defined in Sohn et al., describe a mapping between a character in a Potential Short-form and a character in a Potential Long-form. However, in contrast to Ab3P, Yeganova et al., do not combine these features into hand-crafted strategies. They provide the learner with

---

<sup>3</sup> <http://biotext.berkeley.edu/software.html>

all these features and feature pairs and let the training process weight them. Feature weights are then used to identify abbreviation definitions.

## Converting into BioC

### 1. BioC-compliant modules

We found it straightforward to convert the original software tools for Abbreviation Definition Recognition into BioC-compliant tools. We would also like to point out that the original Schwartz and Hearst software is written in Java, the original Ab3P software is written in C++ and the original NatLab software is written in Perl. As a result, each implementation used a different BioC library; the necessary links were established so that Schwartz and Hearst algorithm could flow seamlessly with the rest of the BioC-Java code, the Ab3P algorithm with the BioC-C++ code, and NatLab with a SWIG-Perl- BioC implementation (10). The BioC-compliant algorithms differ from the originals in these main points:

- The BioC format includes the precise location of annotations in the original text. The original algorithms did not track the location of their recognized abbreviations. Retrofitting this tracking required considerable effort.
- Any text element in a BioC collection is considered a valid input string to inquire for abbreviation definitions. The module accepts BioC-formatted data, and searches for abbreviations regardless of whether text is organized in sentences, passages, or passages of multiple paragraphs. The precise text offsets are produced accordingly.
- The results are produced in BioC format and recognized abbreviations and their definitions can be compared with the output of any other BioC-compliant abbreviation definition recognition tool which uses the same keyfile. This is demonstrated below where we describe the BioC format of three independent abbreviation corpora. All three corpora are provided as input to the three BioC-modules, and results are compared in the Results section.

### 2. BioC-formatted corpora

The first step in preparing a given corpus into BioC format is deciding how to represent the information present in the corpus. Figure 1 illustrates the BioC format for abbreviation annotations that is used in the three abbreviation corpora used for this study. The corpora we considered for this task consist of annotations in the form of *<short-form, long-form>* pairs. To capture this, and make the corpora versatile for other possible biomedical information retrieval studies, we use this markup:

- For annotation elements, the *infol type =ABBR* semantically identifies the annotations as abbreviations. This allows the possibility of having multiple layers of annotations on the same textual data, even including annotations on other entity types that may also overlap. All such annotations can be added without confusion.

- An additional infon element identifies two parts of an abbreviation: ShortForm and LongForm thus preserving the corpora original representation of an abbreviation definition as a *<short-form, long-form>* pair.
- Finally, a relation element reflects the pairing between a short-form and a corresponding long-form. The same infon type “ABBR” is repeated for the relation to make it easier at a semantic level to distinguish what is being annotated and how they relate together.

Documents may contain multiple abbreviation definitions. The location element links the definition to the exact textual coordinates it is mentioned, and also allows for defining a mention composed of multiple con-consecutive substrings. Location information was not present in original annotations, so this is an enrichment over the original versions.

```

<annotation id="SF1014">
  <infon key="type">ABBR</infon>
  <infon key="ABBR">ShortForm</infon>
  <location offset="79" length="2"/>
  <text>FA</text>
</annotation>
<annotation id="LF1014">
  <infon key="type">ABBR</infon>
  <infon key="ABBR">LongForm</infon>
  <location offset="63" length="14"/>
  <text>Fanconi anemia</text>
</annotation>
<relation id="R1014">
  <infon key="type">ABBR</infon>
  <node refid="SF1014" role="ShortForm"/>
  <node refid="LF1014" role="LongForm"/>
</relation>

```

**Figure 2.** Illustration of abbreviation annotation in BioC format.

In order to test the abbreviation definition recognition modules, we converted three abbreviation definition recognition corpora to BioC format. These corpora are: the Ab3P corpus (6), the BIOADI corpus (9) and an earlier version of the Medstract corpus (11). The corpora consist of 1250, 1201 and 199 PubMed citations respectively, and the number of total (and unique) abbreviations annotated in each corpus is: 1223 (1113), 1720 (1491), and 159 (152), respectively, as shown in Table 1. To further highlight the inherent ambiguity in this task, there are 998, 1330, and 146 unique short-forms in the Ab3P, BIOADI and Medstract corpora respectively.

The original versions of these corpora consisted of text files where documents were separated by blank lines. For each document we were given a passage of text (in the Ab3P corpus this was divided into PubMed title and PubMed abstract lines, in the BIOADI corpus title and abstract lines were concatenated together), followed by a list of *<short-form, long-form>* pairs of abbreviations mentioned in the text. PubMed document IDs were given for the Ab3P and

BIOADI corpora, while Medstract documents had Journal citation information and their author list. In converting these corpora to BioC format first we identified PMIDs for all articles, and kept all relevant text for abbreviation definition detection (title and abstract). Next, we identified the correct offsets for each defined abbreviation in the corresponding text. This step included multiple occurrences of each definition within the same passage at times, as well as correct identification of multiple offsets for multiple substrings of some long-form definitions. Naturally, all three corpora went through another step of manual annotation, as many definitions were evaluated and discussed among the four authors. As a result, the final numbers, shown in Table 1, reflect a difference in the number of total annotations per corpus, when compared to the original publications, but we believe that our thorough review has produced better consistency and higher quality corpora.

## Results

We tested the three abbreviation identifying modules on the Ab3P, BIOADI and Medstract corpora as shown in Table 1. Results are based on the new gold standard annotations in the three abbreviation corpora. When compared to the outputs of the algorithms original versions, the BioC-compliant modules produced the same results. The BioC versions, however, have the advantage of being easily combined with other BioC-compliant tools to produce a more complex biomedical text processing system.

**Table 1.** Results of BioC-compliant abbreviation detection modules when tested on BioC-formatted abbreviation corpora.

Corpora	Ab3P	BIOADI	MEDSTRACT
Number of abstracts	1250	1201	199
Number of defined abbreviations	1223	1720	159
Unique number of abbreviations (across the whole corpus)	1113	1421	152
<b>Ab3P Results</b>			
Precision	0.971	0.952	0.993
Recall	0.836	0.770	0.906
F-score	0.898	0.851	0.947
<b>Shwartz&amp;Hearst Results</b>			
Precision	0.950	0.943	0.986
Recall	0.788	0.765	0.893
F-score	0.861	0.844	0.937
<b>NetLab Results</b>			
Precision	0.927	0.853	0.924
Recall	0.879	0.830	0.918
F-score	0.903	0.841	0.921

## Conclusions

We present three easy-to-use, portable, BioC-compatible, interoperable abbreviation definition recognizing modules in biomedical text. The original tools corresponding to Ab3P, Schwartz and Hearst and NatLab algorithms, have only been altered to read and produce the enriched BioC format. The new BioC-compatible modules faithfully preserve their original efficiency, running time power, or other complexity-related aspects, so they can be confidently used as a common pre-processing step for larger multi-layered text-mining endeavors.

We also present three BioC-formatted abbreviation definition recognition corpora that can be used to test the above modules, as well as to study new natural language processing tools. The new versions of the modules, as well as the accompanying corpora, are freely available to the community, through the BioC website: <http://bioc.sourceforge.net>.

## Acknowledgements

This research was supported by the Intramural Research Program of the NIH, National Library of Medicine.

## References

1. Hirschman, L., Yeh, A., Blaschke, C. and Valencia, A. (2005) Overview of BioCreative IV: critical assessment of information extraction for biology. *BMC bioinformatics*, **6 Suppl 1**, S1.
2. Krallinger, M., Morgan, A., Smith, L., Leitner, F., Tanabe, L., Wilbur, J., Hirschman, L. and Valencia, A. (2008) Evaluation of text-mining systems for biology: overview of the Second BioCreative community challenge. *Genome biology*, **9 Suppl 2**, S1.
3. Arighi, C.N., Lu, Z., Krallinger, M., Cohen, K.B., Wilbur, W.J., Valencia, A., Hirschman, L. and Wu, C.H. (2011) Overview of the BioCreative III Workshop. *BMC bioinformatics*, **12 Suppl 8**, S1.
4. Wu, C.H., Arighi, C.N., Cohen, K.B., Hirschman, L., Krallinger, M., Lu, Z., Mattingly, C., Valencia, A., Wieggers, T.C. and John Wilbur, W. (2012) BioCreative-2012 virtual issue. *Database : the journal of biological databases and curation*, **2012**, bas049.
5. Comeau, D.C., Islamaj Doğan, R., Ciccarese, P., Cohen, K.B., Krallinger, M., Leitner, F., Lu, Z., Peng, Y., Rinaldi, F., Torii, M. *et al.* (2013) BioC: a minimalist approach to interoperability for biomedical text processing. *Database*, **2013**.
6. Sohn, S., Comeau, D.C., Kim, W. and Wilbur, W.J. (2008) Abbreviation definition identification based on automatic precision estimates. *BMC bioinformatics*, **9**, 402.
7. Schwartz, A.S. and Hearst, M.A. (2003) A simple algorithm for identifying abbreviation definitions in biomedical text. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, 451-462.
8. Yeganova, L., Comeau, D.C. and Wilbur, W.J. (2011) Machine learning with naturally labeled data for identifying abbreviation definitions. *BMC bioinformatics*, **12 Suppl 3**, S6.
9. Kuo, C.J., Ling, M.H., Lin, K.T. and Hsu, C.N. (2009) BIOADI: a machine learning approach to identifying abbreviations and definitions in biological literature. *BMC bioinformatics*, **10 Suppl 15**, S7.

10. Liu, W., Comeau, D.C., Islamaj Dogan, R. and Wilbur, J. (2013) Extending BioC implementation to more languages.
11. Pustejovsky, J., Castano, J., Cochran, B., Kotecki, M. and Morrell, M. (2001) Automatic extraction of acronym-meaning pairs from MEDLINE databases. *Studies in health technology and informatics*, **84**, 371-375.

# Extending BioC Implementation to More Languages

Wanli Liu, Donald C. Comeau, Rezarta Islamaj Doğan, and W. John Wilbur\*

National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, 8600 Rockville Pike, Bethesda, MD 20894, MD, USA

\*Corresponding author: Tel: 301 435 5926, E-mail: wilbur@ncbi.nlm.nih.gov

## Abstract

As part of a community-wide effort for evaluating text mining and information extraction systems applied to the biomedical domain, BioC is focused on the goal of interoperability: currently a major barrier to wide-scale adoption of text mining tools. BioC is a simple XML format, specified by DTD, for exchanging data for biomedical natural language processing. With initial implementation in C++ and Java, BioC provides libraries of code for reading and writing BioC text documents and annotations. We extend BioC to scripting languages (Perl and Python) with SWIG, as well as to the Go language. BioC modules are functional in these languages, which can facilitate some BioCreative tasks. We also discuss the addition of new languages to support BioC in the future. BioC implementations are freely available at the BioC site: <http://bioc.sourceforge.net>.

## Introduction

BioCreative Workshops provide the forum for text mining, computational linguistics and natural language processing researchers to build, adapt and/or integrate information extraction systems that address biologically meaningful tasks and that provide results of practical relevance. In order to ensure satisfactory community assessment and method comparison, and to promote scientific progress, it is necessary to establish common standards and shared criteria that enable comparison and integration of different approaches. BioC (1) has been gaining momentum as a solution to the interoperability challenge — an interchange format for biomedical natural language processing tools. Expressed in a simple XML format, specified by DTD, text documents and related data annotations can be shared easily between different text mining and information extraction systems applied to the biomedical domain. Moreover, all tools that communicate with this shared format can potentially be combined as parts of larger, more complicated systems.

Considering the variety of computational tools employed by the biomedical text mining community, successful interoperability requires uniform BioC support across various programming language environments. The first releases of BioC code were implemented in C++ and Java, two mainstream programming languages which provide a solid foundation for defining

BioC functionality. However, scripting languages such as Perl and Python have become popular in the communities of bioinformatics and natural language processing for their ease of use with reasonable performance. For example, BioPerl<sup>1</sup> and Biopython<sup>2</sup> are widely adopted for the tasks of parsing BLAST output and querying the GENBANK database (2,3). Go<sup>3</sup>, a newly-emerging language from Google, is also receiving attention with biogo<sup>4</sup> targeted at computationally intensive Bioinformatics tasks. For biomedical natural language processing tasks, a variety of NLP packages (CPAN-NLP<sup>5</sup>, NLTK<sup>6</sup>) have been developed with Perl and Python. Therefore, it is important to make BioC functionalities easily accessible for applications coded with these languages. With the support of BioC modules, applications are able to conveniently extract information from BioC XML files, process the information, and write the output in BioC XML format. This facilitates efficient and uniform data sharing. Directly implementing BioC in a large number of other languages would require a large amount of work and additional exhaustive testing. To efficiently deliver behavior identical to C++ in scripting languages, we employ the interface compiler SWIG<sup>7</sup> to connect the C++ BioC implementation to the target languages: Perl and Python. By supporting full BioC functionality faithfully, the integration of the C++ BioC implementation and scripting languages enables fast and flexible prototyping while relying on the low-level C++ code, to duplicate the behavior of C++ applications. In this paper we introduce the effort of our team to enable BioC users to take advantage of BioC utilities in scripting language applications with some examples explained in detail, as well as our experiments with the Go language.

## BioC Interface with Scripting Languages

To reach high interoperability and reusability in NLP and text processing tasks, BioC is designed as a simple workflow based on XML formats (1). As in the C++ and Java environments, the BioC workflow in a scripting language context uses two connector modules for XML input/output, as shown in Figure 1. In Perl or Python, an Input Connector object is created to gain access to XML input from a file or network stream. This input is converted to data encapsulated in BioC data classes. The BioC data classes then provide various methods to retrieve data for the data processing stage, which is implemented in a scripting language. Modify methods of BioC data classes can be used to update BioC data classes, or new data can be created. Finally, the output connector writes the new or updated data in XML format.

As shown in Figure 1, both BioC data classes and Input/Output connectors are coded in C++, while the data processing stage is coded in a scripting language. Data travels between C++ and

---

<sup>1</sup> <http://www.bioperl.org>

<sup>2</sup> <http://biopython.org/>

<sup>3</sup> <http://golang.org/>

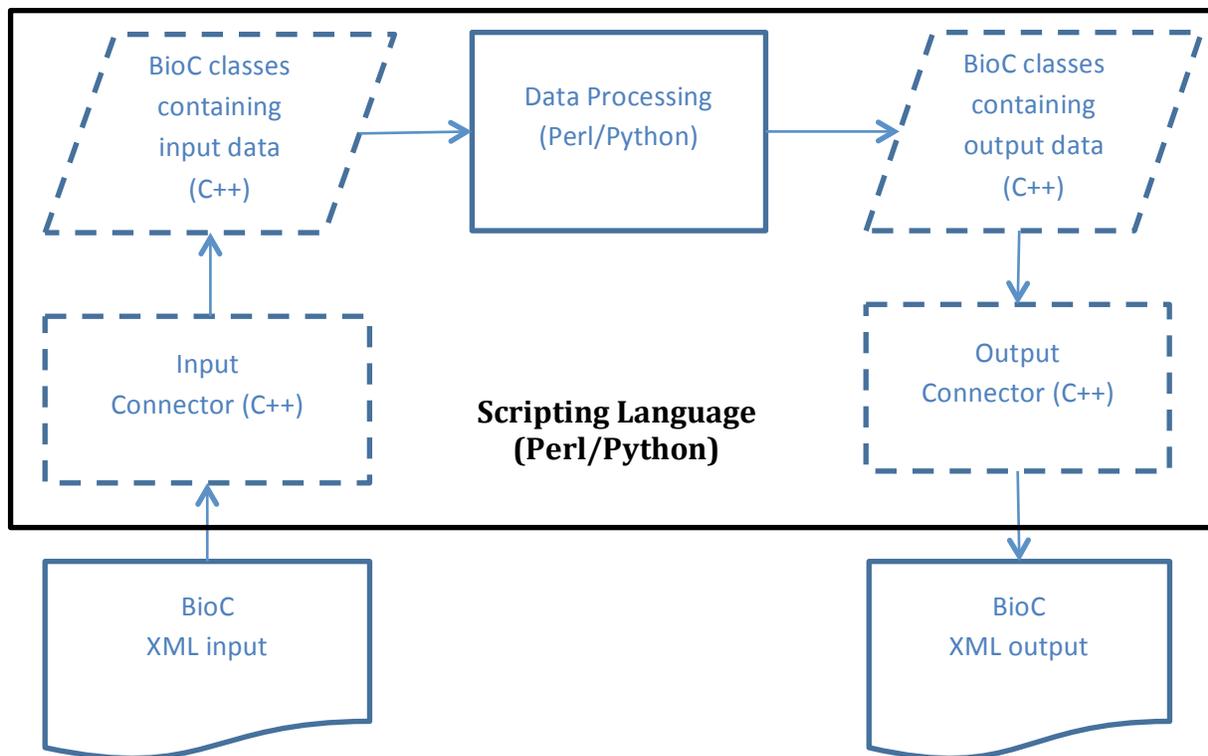
<sup>4</sup> <https://code.google.com/p/biogo/>

<sup>5</sup> <http://cpan.org>

<sup>6</sup> <http://nltk.org>

<sup>7</sup> <http://www.swig.org>

scripting language contexts through BioC class methods. The separation of the data processing stage and the Input/Output connectors enables the data processing stage to focus on processing biomedical data, regardless of the XML format of the Input/Output files. As in C++ applications, input stage, data processing stage, and output stage can be decoupled as needed.



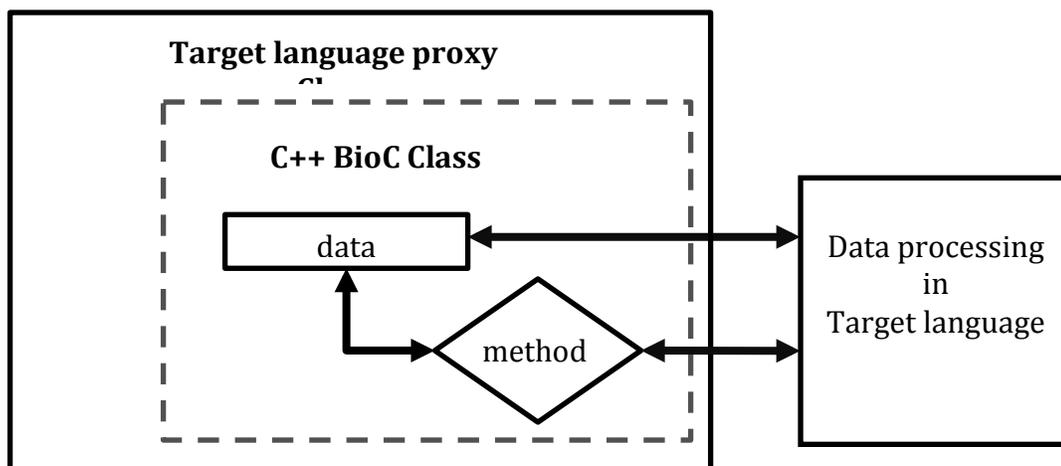
**Figure 1.** BioC workflow for scripting language application

### A BioC C++ SWIG Interface

To obtain seamless data flow in BioC, it is critical to initialize Input/Output connectors, and communicate via methods of BioC classes in the target languages. To accomplish this, the BioC C++ header files (*BioC.hpp*, *BioC\_libxml.hpp*, *BioC\_util.hpp*) are processed by SWIG, which extracts the declarations in these header files to create wrapper codes in both the target scripting language and C++. The wrapper code in the target language defines proxy classes for the underlying C++ classes, as well as a variety of customization to suit the specific target language features. The original BioC source code is compiled by a C++ compiler to generate object file, in the same way as building a pure C++ application. These BioC objects are linked with an object compiled from C++ wrapper code to provide BioC proxy classes for the target language. The proxy classes can be accessed in the target language to provide methods to manipulate the data contained in the C++ BioC classes, as shown in Figure 2.

## Using BioC in Scripting Languages

To illustrate the key points of using a BioC module in target languages, we use Perl code as an example (Figure 3). Based on BioC C++ version 1.0<sup>8</sup>, SWIG version 2.0.4<sup>9</sup> produces BioC\_Perl.so and BioC\_Perl.pm. BioC\_Perl.pm contains the Perl proxy classes for the BioC C++ classes, and BioC\_Perl.so provides the executable implementation of BioC classes.



**Figure 2.** Access C++ BioC Class through target language proxy class wrapper interface

After importing the Perl BioC module, we are ready to initialize BioC wrapper objects. The Connector\_libxml class object (`$xml`) enables opening an XML file and reading the contained data into a Collection class object (`$collection`). Then we can iterate through the BioC XML file by visiting each document (`$dcm`) within the collection, and each passage (`$psg`) within each document, and each annotation (`$ann`) and relation (`$rel`) within each passage. While iterating through the XML file, the data members defined in BioC classes (e.g., `{id}` of Document class) can also be accessed directly or through methods via Perl wrapper objects. In addition to reading the whole XML file into (`$collection`) prior to processing, BioC also provides a `read_next(document)` method for one-document-at-a-time access.

In addition to the data read into the BioC classes, new data can be added to the BioC classes. One such example is the `{infons}` data structure in a number of BioC classes, which is a C++ `std::map` container template mapping a key string to a value string. The mapped value string can be retrieved via the `get()` method of the `{infons}` data structure with `ELEMENT KEY string` (and updated via the `set()` method).

After data is extracted from XML format and then processed by Perl code, the original collection class object can be modified in a fashion similar to reading BioC data classes. The updated

<sup>8</sup> [http://sourceforge.net/projects/bioc/files/BioC\\_C%2B%2B\\_1.0.tar.gz/download](http://sourceforge.net/projects/bioc/files/BioC_C%2B%2B_1.0.tar.gz/download)

<sup>9</sup> <http://sourceforge.net/projects/swig/files/swig/swig-2.0.4/>

collection class object can be saved in XML format, ready to be accessed by another BioC application. The Perl BioC module has been successfully used by the NatLab abbreviation system (4) to make it BioC compatible (5).

```
# import BioC module from PATH TO PERL BIOC MODULE, where
# BioC_Perl.pm and BioC_Perl.so are located
BEGIN {push (@INC, PATH TO PERL BIOC MODULE);}
use BioC_Perl;

my $collection = new BioC_Perl::Collection();
my $xml = new BioC_Perl::Connector_libxml();
$xml->read (XML INPUT, $collection);

# iterate through all documents
for ( my $i = 0; $i < $collection->{documents}->size(); $i++) {
    my $dcm = $collection->{documents}->get($i);
    print "$dcm->{id}\n";
    print "$dcm->{infons}->get(ELEMENT KEY)\n";
    # iterate through all passages
    for ( my $j = 0; $j < $dcm->{passages}->size(); $j++) {
        my $psg = $dcm->{passages}->get($j);
        # iterate through all annotations
        for ( my $k = 0; $k < $psg->{annotations}->size(); $k++) {
            my $ann = $psg->{annotations}->get($k);
        }
        # iterate through all relations
        for (my $k = 0; $k < $psg->{relations}->size(); $k++) {
            my $rel = $psg->{relations}->get($k);
        }
    }
}
... # process data and modify $collection if necessary
$xml->write (XML OUTPUT, $collection);
```

**Figure 3.** Perl code accessing BioC module (tested with Perl 5.8.8)

Figure 4 shows Python code executing the same task as Perl code. Compared to Perl, BioC data classes interface with native Python structures more naturally, which improves convenience and ease of development. Unlike in Perl, the C++ `std::map` container is treated as a Python `dict` object, and a mapped string can be accessed as `dict[key]`. The example code demonstrates several ways to loop over BioC data. However, the stronger support in Python is still not complete. SWIG has not implemented all Python methods for the underlying C++ data structures. For example, the `get()` method returning the default value for a missing key in a Python `dict` object is not available in the SWIG implementation (although one can work around this particular issue by using the key in `dict` expression, which is supported). Another pitfall in applying our Perl and Python BioC modules has to do with the way the underlying C++ data is accessed. When updating the BioC data, references should be used in order to commit the new value (as shown in the code examples). SWIG sometimes implicitly

make a copy of the C++ data, and as a result, the updates or changes to the Python or Perl copy of the data are not reflected in the C++ data and may be lost. Care must be taken when updating or changing data to ensure the original C++ data is changed. A better practice may be to create new data objects with updated or new data when memory capacity permits.

```
# import BioC module from PATH TO PYTHON BIOC MODULE, where
# BioC_Python.py and _BioC_Python.so are located
sys.path.append(PATH TO PYTHON BIOC MODULE)
import BioC_Python

collection = BioC_Python.Collection()
xml = BioC_Python.Connector_libxml()
xml.read (XML INPUT, collection)

# iterate through all docuemnts with iterator dcm          (read only)
for dcm in collection.documents :
    print dcm.id
    print dcm.infons[ELEMENT KEY]
    # iterate through all passages
    for index, psg in dcm.passages:
        # iterate through all annotations with reference
        # (read/write)
        for k in range(0, psg.annotations.size()):
            ann = psg.annotations[k]

        # iterate through all relations with iterator (read only)
        for rel in psg.relations:

... # process data and modify $collection if necessary
xml.write (XML OUTPUT, collection)
```

**Figure 4.** Python code accessing BioC module (tested with Python 2.5.1)

## A BioC Go Implementation

Go is a new language from Google developed by Ken Thompson and Rob Pike, who are known for UNIX, C, and Plan 9. Its features include the convenience of type inference and the goroutines for concurrency. As their web page<sup>10</sup> says, “It’s a fast, statically typed, compiled language that feels like a dynamically typed, interpreted language.”

In Go, XML data can be marshaled and unmarshaled quite simply by using struct tags. One limitation is that there is no direct way to exchange information between XML and a map, as used to implement infons in other languages. So Go BioC objects hold both an Infons map and an InfonStructs slice. The XML data is marshaled and unmarshaled out of and in to the InfonStructs slice. The data is moved in to or out of the Infons map after reading or before writing. This is a small inefficiency because the amount of infon data is small. Reading or

---

<sup>10</sup> <http://golang.org/doc/>

writing a BioC collection a document at a time does require lower level interaction with the XML parser for the collection itself. The individual documents can still be written and read with the direct Marshal and Unmarshal functions. So far we are pleased with the language and expect to continue our experiments.

## Conclusions

We describe the extension of BioC beyond C++ and Java programming languages, enabling convenient handling of BioC XML documents in other languages. SWIG allows the original behavior of C++ BioC classes to be available in scripting language contexts by reusing the same C++ source codes. However, different languages have different features and capabilities, which are exposed to varying extents by SWIG<sup>11</sup>. Python is demonstrated to offer a much richer interaction with the C++ data structures than is available in Perl. Future versions of SWIG may provide more support for additional language features. A native implementation provides more natural interaction and integration with the language. However, it requires more effort than using SWIG and requires testing to guarantee consistent behavior with other BioC implementations. We are interested in suggestions from BioC users regarding additional languages, such as Ruby, and new programming features. Perl, Python, and Go BioC implementations are freely available at the BioC website.

## Acknowledgements

This research was supported by the Intramural Research Program of the National Institutes of Health, National Library of Medicine.

## References

1. Comeau, D.C., Islamaj Doğan, R., Ciccarese, P., Cohen, K.B., Krallinger, M., Leitner, F., Lu, Z., Peng, Y., Rinaldi, F., Torii, M. *et al.* (2013) BioC: a minimalist approach to interoperability for biomedical text processing. *Database*, **2013**.
2. Stajich, J.E., Block, D., Boulez, K., Brenner, S.E., Chervitz, S.A., Dagdigian, C., Fuellen, G., Gilbert, J.G.R., Korf, I., Lapp, H. *et al.* (2002) The Bioperl Toolkit: Perl Modules for the Life Sciences. *Genome Research*, **12**, 1611-1618.
3. Cock, P.J.A., Antao, T., Chang, J.T., Chapman, B.A., Cox, C.J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B. *et al.* (2009) Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, **25**, 1422-1423.
4. Yeganova, L., Comeau, D.C. and Wilbur, W.J. (2011) Machine learning with naturally labeled data for identifying abbreviation definitions. *BMC bioinformatics*, **12 Suppl 3**, S6.
5. Islamaj Dogan, R., Comeau, D.C., Yeganova, L. and Wilbur, J. (2013) Finding abbreviations in biomedical literature: Three BioC-compatible modules and three BioC formatted corpora.

---

<sup>11</sup> <http://www.swig.org/Doc2.0/>

# Natural Language Processing Pipelines to Annotate BioC Collections with an Application to the NCBI Disease Corpus

Donald C. Comeau<sup>\*</sup>, Haibin Liu, Rezarta Islamaj Doğan and W. John Wilbur

National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, 8600 Rockville Pike, Bethesda, MD 20894, MD, USA

<sup>\*</sup>Corresponding author: Tel:301-435-5887, E-mail: comeau@ncbi.nlm.nih.gov.

## Abstract

We have implemented BioC natural language preprocessing pipelines in two popular programming languages: C++ and Java. They are largely based on the well-known natural language processing tool sets, MedPost and Stanford. Tools integrated in the pipelines include sentence segmentation, tokenization, part-of-speech (POS) tagging, lemmatization and sentence parsing. These pipelines can be easily integrated along with other BioC programs into new BioC compliant text mining systems. We converted the NCBI disease corpus to BioC format and all the tools described here were run on this corpus to demonstrate their functionality. Code and data can be downloaded from: <http://bioc.sourceforge.net>.

## Introduction

BioC (1) is a new format and associated source code libraries for sharing text and annotations. This allows for the simple and convenient processing of text corpora. With the provided libraries, it is straightforward to incorporate BioC code into existing programs to read in data from BioC formatted input files and write out results to BioC formatted output files.

Text preprocessing is integral to virtually all natural language processing (NLP) systems. It reformats the original text into meaningful units that contain important linguistic features before performing subsequent text mining strategies. Generally, several preprocessing steps need to be performed, such as sentence segmentation, part-of-speech (POS) tagging, and sentence parsing. Poor text preprocessing performance will have a detrimental effect on downstream processing. Compared to general English texts, a particularly challenging aspect of preprocessing biomedical text is the wide variety of domain-specific technical terminology encountered.

Our contribution to the interoperability track of the BioCreative IV challenge is BioC text-preprocessing pipelines in C++ and Java. These tools integrate a selection of state-of-the-art text preprocessing tools and produce corresponding text analyses in the BioC XML format. The integrated tools are considered representative in the target domains and have been reported to

yield competitive results on biomedical texts. Instead of being all-inclusive, the intension of our work is to provide essential text preprocessing functionalities to BioC users. The processing is implemented in a flexible way so that users can incorporate other tools according to their needs. The implementation is freely available to the NLP and text mining research communities, and is released as open source software that can be downloaded.

While many researchers have their own favorite natural language preprocessing tools, it is useful to have examples of commonly used tools available in the BioC format. We use the NCBI disease corpus as a model corpus. The outputs of these programs provide examples of how the BioC format links the results of different tools in an interoperable and integrated fashion. This demonstrates how different programs can use and produce data in a consistent format, regardless of their implementation language.

## **C++**

C++ is a high performance, compiled language with very good execution time and memory usage performance. Figure 1 shows the overall flow of our pipeline. The different tools, sentence segmentation, tokenization, part-of-speech tagging, and dependency parsing, are implemented as separate stand-alone programs. They are represented by the inner boxes in the Figure 1. This is convenient if the results of only one, or a few, of the tools are desired.

The pieces of the C++ pipeline were drawn from the MedPost (2) collection of natural language processing tools. One challenge was that MedPost normalizes some of the results. For example, multiple spaces between tokens in a sentence would be normalized to a single space. This had to be taken into account when determining the offset to the original text as encouraged by the BioC format.

### **Sentence Segmenting and Tokenizing**

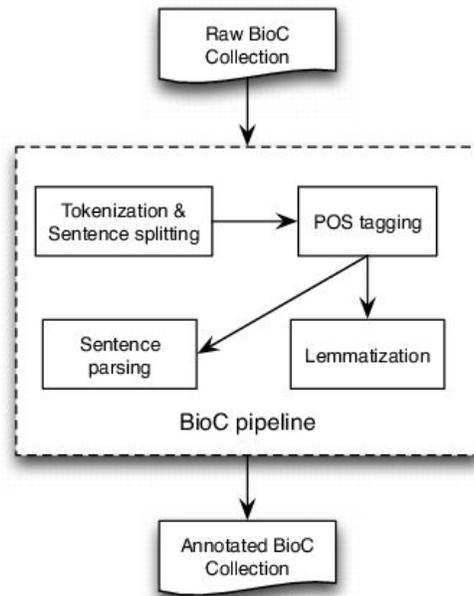
MedPost uses a rule based approach to sentence segmenting. A recent paper using a token lattice design pattern and the adapted Viterbi algorithm achieved a tokenization accuracy of 92.4% compared to MedPost's 92.9% (3).

### **Parts-of-Speech**

The principle MedPost tool is a high accuracy part-of-speech tagger trained on a MEDLINE corpus (2). Using its own tag set, it achieves 97.43% accuracy on a 1000 sentence test set. It achieves 96.9% accuracy using the Penn Treebank tag set. In addition to part-of-speech, MedPost can tokenize text and segment sentences. It has been widely used and is included in the popular LingPipe natural language toolkit<sup>1</sup>. While better results have been achieved on a specialized medical sublanguage, MedPost still provided a valuable baseline (4).

---

<sup>1</sup> <http://alias-i.com/lingpipe/>



**Figure 1 - BioC Text-preprocessing Pipeline**

### **Lemmatization**

The C++ pipeline does not include its own lemmatizer. For the BioNLP 2013 shared task, the C++ pipeline was run to prepare supporting material (6). The Java Biolemmatizer (5) was also run, to complete our supporting material package. This demonstrates a benefit of a language neutral data format. Results can be easily combined regardless of the implementation language.

### **Dependency Parse**

MedPost also includes a wrapping of the C&C dependency parser (7). In addition to the expected head and dependent tokens, some relations include a type token. This is described in the `cnc.key` file.

### **Java**

Figure 1 also presents the detailed annotation flow of the Java implementation of our BioC text-preprocessing pipeline, which includes text tokenization, sentence segmenting, POS tagging, lemmatization and sentence parsing. The pipeline takes as input a BioC collection. Preprocessing is then invoked for each BioC passage on which the integrated tools are performed sequentially to produce corresponding text analyses. In the end, the generated annotations, along with the BioC collection information, are inserted into a BioC data structure to produce an annotated BioC XML file.

### **Sentence Segmenter**

An efficient sentence segmenter, DocumentPreprocessor, is used to produce a list of sentences from a plain text. It is a creation of the Stanford NLP group using a heuristic finite-state machine

that assumes the sentence ending is always signaled by a fixed set of characters. Tokenization is performed by the default, rule-based tokenizer of the sentence segmenter, PTBTokenizer, prior to the segmenting process in order to divide text into a sequence of tokens. The “invertible” option of the tokenizer is invoked to ensure that multiple whitespaces are reflected in token offsets so that the resulting tokens can be faithfully converted back to the original text. Sentence segmentation is then a deterministic consequence of tokenization.

### **POS tagging**

The MaxentTagger based on a maximum entropy model is used for part-of-speech tagging. The MaxentTagger is also the default POS tagger used by the Stanford parser before parsing the text.

### **Lemmatization**

BioLemmatizer (5) is used to perform the morphological analysis of biomedical literature. It has been demonstrated that the BioLemmatizer achieves the best lemmatization performance on biomedical texts and contributes to biomedical information retrieval/extraction tasks. The word form and the part-of-speech of a token are required as input to the BioLemmatizer to retrieve the corresponding lemma.

### **Sentence parsing**

The POS-tagged sentences are then submitted to the Stanford unlexicalized natural language parser (8) to analyze the syntactic and semantic structure of the sentences. The Stanford parser has been reported to be one of the state-of-the-art parsers in terms of speed and accuracy (9,10). When applied to the biomedical domain, it has successfully helped to extract various types of biological relations and events from the literature (11,12) and identify medical treatment terms from randomized clinical trial (RCT) reports (13). The Stanford parser is parameterized to return both Penn Treebank parse tree and dependency representations for each sentence. While the flat version of the Penn Treebank parse tree is directly encoded into the XML, the dependency representations are recorded directly in BioC as grammatical relations between participating tokens referred to by their token IDs.

Because of the Unicode compatibility of the integrated tools, the pipeline should work well over texts encoded in both ASCII and Unicode. The pipeline currently performs an end-to-end annotation from text tokenization to sentence parsing. However, even though sentence parsing is useful for tasks such as question answering or relation extraction, it is not often considered by tasks like named entity detection or concept recognition. In addition, due to the constituent-based parsing nature of the Stanford parser, sentence parsing accounts for most of the execution time of the pipeline. Therefore, we plan to provide more flexibility to the pipeline users in the next release to allow them to choose the annotation steps according to their needs.

### **Comparison**

We compared the output of the C++ and Java BioC tools by running them on a small set of 10 PubMed references. We performed a detailed comparison of the output of both pipelines. The

BioC-formatted output could be easily handled by the same BioC-compatible program. The segmented sentences were identical, as expected, since the set of abstracts was known not to have any challenging cases. The tokens and parts-of-speech tags were very similar, with only expected differences. For example, MedPost and Stanford parsers make different decisions on splitting tokens containing hyphens (-) or slash (/) characters.

The dependency graphs were verified by processing them with the same program to produce visual graphs based on the graph description language DOT (14). Again, using the same program to produce graphs is a benefit of taking advantage of the BioC format for the output of both pipelines.

### **Application of BioC NLP tools on the NCBI disease corpus**

We used the NCBI disease corpus to provide a more rigorous evaluation of the BioC NLP tools. The NCBI disease corpus (15,16) is a manually annotated resource for disease name recognition and normalization in biomedical text, which comprises a collection of 793 PubMed abstracts and a total of 6,892 disease mentions, which further correspond to 790 disease concepts mapped to MeSH descriptors or OMIM identifiers. It was completed by a team of fourteen annotators in three annotation rounds and provides a high-quality, reliable and consistently annotated resource. This resource was used to develop a highly effective disease normalization method (17). In order to make the NCBI disease corpus more accessible, and to promote its usage for other related biomedical information extraction tasks, the collection was converted to BioC-XML format, and is used here as a model test case to run the BioC NLP pre-processing pipeline tools.

Figure 2 illustrates the disease mention and concept annotation in the NCBI disease corpus expressed in BioC XML format. Each annotation contains the textual mention with the appropriate location information, given with the precise document offset and length. Since the annotation of the same textual string is two-faceted, infons are used to express the semantics of the annotation: the infon key="EntityType" is used to distinguish the four disease categories as specified by the annotators of the corpus, the infon key="Nomenclature" is used to distinguish the correct terminology resource selected for the annotation, and the infon key="ConceptID" specifies the unique concept identifier for the textual mention. The infon key-value pairs in the annotation elements correspond to the original corpus format, the PubTator format (18). The tool to convert PubTator annotation data to BioC is also available for download.

Next, we ran both C++ and Java pipelines on the BioC-formatted NCBI disease corpus, thereby enriching this resource with machine-assisted annotations and basically pre-processing the data ready for use by any BioC compliant application. The machine-assisted annotations consist of: sentence segmentation, tokenization and POS tagging processed using both MedPost and Stanford parsers, lemmatization using BioLemmatizer, dependency parsing using both C&C and

Stanford parses, as well as abbreviation definition detection using Ab3P, Schwartz & Hearst and NatLab algorithms(19).

```
Familial deficiency of the seventh component of complement associated with
recurrent bacteremic infections due to Neisseria.

<annotation id = "D0">
  <infony key="type">Disease</infony>
  <infony key="EntityType">SpecificDisease</infony>
  <infony key="Nomenclature">OMIM</infony>
  <infony key="ConceptID">610102</infony>
  <location offset="1" length ="58"/>
  <text>Familial deficiency of the seventh component of
complement</text>
</annotation>
<annotation id = "D1">
  <infony key="type">Disease</infony>
  <infony key="EntityType">DiseaseClass</infony>
  <infony key="Nomenclature">MeSH</infony>
  <infony key="ConceptID">D016870</infony>
  <location offset="86" length ="38"/>
  <text>bacteremic infections due to Neisseria</text>
</annotation>

<annotation id="0">
  <infony key="type">token</infony>
  <infony key="POS">JJ</infony>
  <infony key="lemma">familial</infony>
  <location offset="0" length="8"/>
  <text>Familial</text>
</annotation>
<annotation id="1">
  <infony key="type">token</infony>
  <infony key="POS">NN</infony>
  <infony key="lemma">deficiency</infony>
  <location offset="9" length="10"/>
  <text>deficiency</text>
</annotation>

<relation id="R0">
  <infony key="relation">amod</infony>
  <node refid="1" role="head"/>
  <node refid="0" role="dependent"/>
</relation>
```

**Figure 2.** Illustration of annotations in the enriched NCBI disease corpus, manual annotations of disease mentions and concepts, and BioC-tools produced annotations for text pre-processing.

## Summary

We have implemented BioC natural language preprocessing pipelines in two popular programming languages: C++ and Java. They are largely based on well-known natural language processing tool sets, MedPost and Stanford. A benefit of BioC is the interoperability between tools written in different programming languages. Using these tools, it is straightforward to use the Stanford tools in a C++ pipeline, or MedPost in a Java pipeline.

The BioC text preprocessing pipelines serve as starting points for automatically annotating BioC collections. The pipelines are implemented in a flexible manner enabling researchers to integrate other tools as needed. Alleviating the burden of interoperability challenges will encourage the development of novel approaches allowing improved natural language processing performance. We also present the NCBI disease corpus in the BioC XML format. As a test case application for the pipelines, the corpus is used as input to all BioC NLP tools in both C++ and Java. The result is a rich collection of annotations that combines manual annotations (for disease entity mentions and concept normalization) with tool-provided machine annotations. This collection, along with the BioC tools, is available to the community from the BioC website: <http://bioc.sourceforge.net>.

## Acknowledgements

This research was supported by the Intramural Research Program of the NIH, National Library of Medicine.

## References

1. Comeau, D.C., Islamaj Doğan, R., Ciccarese, P., Cohen, K.B., Krallinger, M., Leitner, F., Lu, Z., Peng, Y., Rinaldi, F., Torii, M. *et al.* (2013) BioC: a minimalist approach to interoperability for biomedical text processing. *Database*, **2013**.
2. Smith, L., Rindfleisch, T. and Wilbur, W.J. (2004) MedPost: a part-of-speech tagger for bioMedical text. *Bioinformatics*, **20**, 2320-2321.
3. Barrett, N. and Weber-Jahnke, J. (2011) Building a biomedical tokenizer using the token lattice design pattern and the adapted Viterbi algorithm. *BMC bioinformatics*, **12 Suppl 3**, S1.
4. Liu, K., Chapman, W., Hwa, R. and Crowley, R.S. (2007) Heuristic sample selection to minimize reference standard training set for a part-of-speech tagger. *Journal of the American Medical Informatics Association : JAMIA*, **14**, 641-650.
5. Liu, H., Christiansen, T., Baumgartner, W.A., Jr. and Verspoor, K. (2012) BioLemmatizer: a lemmatization tool for morphological processing of biomedical text. *Journal of biomedical semantics*, **3**, 3.
6. Stenetorp, P., Golik, W., Hamon, T., Comeau, D.C., Islamaj Dogan, R., Liu, H. and Wilbur, W.J. (2013) BioNLP Shared Task 2013: Supporting Resources. *Proceedings of the BioNLP Shared Task 2013 Workshop*, 99--103.
7. Clark, S. and Curran, J.R. (2004) Parsing the WSJ using CCG and log-linear models. *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, 103.

8. Klein, D. and Manning, C.D. (2003) Accurate unlexicalized parsing. *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, 423-430.
9. Hempelmann, C.F., Rus, V., Graesser, A.C. and McNamara, D.S. (2005) Evaluating state-of-the-art treebank-style parsers for Coh-metrix and other learning technology environments. *Proceedings of the second workshop on Building Educational Applications Using NLP*, 69-76.
10. Cer, D.M., de Marneffe, M.-C., Jurafsky, D. and Manning, C.D. (2010) Parsing to Stanford Dependencies: Trade-offs between Speed and Accuracy. *LREC*.
11. Miyao, Y., Sagae, K., Sætne, R., Matsuzaki, T. and Tsujii, J.i. (2009) Evaluating contributions of natural language parsers to protein-protein interaction extraction. *Bioinformatics*, **25**, 394-400.
12. McClosky, D., Surdeanu, M. and Manning, C.D. (2011) Event extraction as dependency parsing. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, 1626-1635.
13. Xu, R., Morgan, A., Das, A.K. and Garber, A. (2009) Investigation of unsupervised pattern learning techniques for bootstrap construction of a medical treatment lexicon. *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*, 63-70.
14. Ellson, J., Gansner, E.R., Koutsofios, E., North, S.C. and Woodhull, G. (2004) In Junger, M. and Mutzel, P. (eds.), *Graph Drawing Software*. Springer-Verlag, Berlin/Heidelberg, pp. 127--148.
15. Islamaj Dogan, R. and Lu, Z. (2012) An improved corpus of disease mentions in PubMed citations. *Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*, 91-99.
16. Islamaj Dogan, R., Leaman, R. and Lu, Z. (2013) NCBI Disease Corpus: A Resource for Disease Name Recognition and Concept Normalization. *submitted*.
17. Leaman, R., Islamaj Dogan, R. and Lu, Z. (2013) DNorm: Disease Name Normalization with Pairwise Learning to Rank. *Bioinformatics*.
18. Wei, C.H., Kao, H.Y. and Lu, Z. (2013) PubTator: a web-based text mining tool for assisting biocuration. *Nucleic Acids Res*, **41**, W518-522.

# Brat2BioC: conversion tool between brat and BioC

Antonio Jimeno Yepes<sup>1,2</sup>, Mariana Neves<sup>3,4</sup>, Karin Verspoor<sup>1,2</sup>

<sup>1</sup>NICTA Victoria Research Lab, Melbourne VIC 3010, Australia

<sup>2</sup>Department of Computing and Information Systems, University of Melbourne, Melbourne VIC 3010, Australia

<sup>3</sup>Humboldt-Universität zu Berlin, WBI, Berlin, Germany

<sup>4</sup>Berlin Brandenburg Center for Regenerative Therapies, Charité, Berlin, Germany

## Introduction

Interoperability between text mining solutions requires sharing information, specifically resources such as annotated corpora, in a common format. Several formats are available that have been used in the biomedical natural language processing (BioNLP) community, though no single standard has emerged. The BioC formalism [1] is intended to fill this gap, by providing tools to work with BioC, in addition to the proposed format itself. Translation of annotations of commonly used formats into BioC allows reusing existing annotated corpora with BioC solutions. The standoff *brat* (brat rapid annotation tool) format<sup>1</sup> is one of the more commonly used formats. For instance it has been used in the BioNLP shared task series [2]. Several corpora have been made available in the brat format, including the Human Variome Project corpus<sup>2</sup> and the CellFinder corpus<sup>3</sup>[3]. We have prepared a software solution, named Brat2BioC, that translates annotations originally in brat format into BioC and vice versa. The Brat2BioC tool is available in bitbucket at [https://bitbucket.org/nicta\\_biomed/brat2bioc](https://bitbucket.org/nicta_biomed/brat2bioc).

## Methods

The Brat2BioC tool was developed in the Java programming language, using provided BioC code<sup>4</sup> to model the data using BioC objects and to serialize and deserialize BioC files.

Several differences exist between the two formats. These include the physical division of data and annotations among various files, and the representational choices for entity and relation annotations. These differences need to be resolved in order to perform the mapping between the two formats.

---

<sup>1</sup> Brat standoff annotation: <http://brat.nlplab.org/standoff.html>

<sup>2</sup> Human Variome Project corpus: <http://www.opennicta.com/home/health/variome>

<sup>3</sup> CellFinder corpus: <http://www.informatik.hu-berlin.de/forschung/gebiete/wbi/resources/cellfinder>

<sup>4</sup> BioC java: [http://www.ncbi.nlm.nih.gov/CBBresearch/Dogan/BioC/BioC\\_Java\\_1.0.tar.gz](http://www.ncbi.nlm.nih.gov/CBBresearch/Dogan/BioC/BioC_Java_1.0.tar.gz)

## File representation

The brat format for annotated documents assumes that the raw text of documents appears in one file and annotations associated with that raw text appear in a separate file or files. Typically, one file is provided for each document and several files are provided for the annotations. On the other hand, BioC can handle annotations of several documents and document passages within the same file. In our implementation, the set of document files from the source brat files are converted to a single BioC file.

For an annotated corpus in BioC, all documents and annotations can be integrated into the same file, while the brat format requires a file for each document text (“\*.txt”) and one file for the annotations on a given document (“\*.ann”); for the BioNLP Shared Tasks there are typically two annotation files, “\*.a1” and “\*.a2”). An additional difference is that brat has no explicit mechanism for representing internal document structure. In most existing uses of brat, a single source text is divided into several smaller files, each corresponding to a section of the source document. The name of those files typically is used to convey the meta-data about the source document and the section of that document that the file corresponds to. For instance the file name “2265717-01-Abstract-p01.txt” indicates that the file contains the first paragraph in the abstract of the document with PubMed identifier (PMID) 2265717. In some cases, as in the BioNLP shared task 2009, the name of the file is just the PMID, indicating that the file contains the title and abstract text associated with that PMID.

As mentioned above, file extensions in brat indicate the type of data in a file. In our BioC conversion, we capture this information through an *infon* object that specifies the extension of the source file in which the annotation was found (*a1*, *a2* or *ann*). The implementation offered by the BioC C++ code approaches this by generating several files, but we have preferred a more compact approach to the problem, thus requiring just one file to be generated.

When converting a BioC file into brat files, the extension of the annotation file(s) should be provided. If the extension information is not provided in the BioC file, by default the annotations are added to a file that is given a name corresponding to the value of the *id* tag of the document, and with default extension *ann*.

In our mapping from brat to BioC, the BioC document *id* tag is set to the name of the brat source file without the extension. This is a convention commonly used in several shared tasks and annotation efforts using the brat format. The document text, in txt brat files, is entered as a single passage tag in the BioC format. No assumptions are made about the intrinsic structure of the text documents since this structure is not defined in the brat format. An example of the high-level structure of a brat document mapped to BioC format is presented below in Figure 1:

```
<document><id>2265717-01-Abstract-p01</id><passage><offset>0</offset> <text>**  
IGNORE LINE **...</text>
```

**Figure 1.** Document text example

### **Conversion of different types of annotations from brat to BioC**

Information provided by the brat format can be mapped into the BioC representation due to its flexibility but this flexibility implies that there are some BioC features that are not available in the brat format. In this section, we explain the conversion decisions for annotation types that are explored and some examples are provided. Brat has several annotation types that have been modelled as BioCAnnotation and BioCRelation objects. In brat, the type of annotation is denoted by the first letter of the first token denoting as well the identifier. The same notation is used to denote the different type of annotations as in BioC. The identifier from the brat file is considered as the identifier of the BioC object.

We have compared our initial conversion proposal with the one proposed by Yifan Peng, Vijay Shanker and Cathy Wu [4], used in their iSimp tool<sup>5</sup>. We found several differences. The first one is that they separate a brat document text into different passages according to newlines, while we just enter the text in a single *passage* tag. The second is that we initially used an *infol* tag to store an event trigger instead of storing it in a *node* tag. We adjusted our proposal to store the event trigger in a *node* tag, since the event trigger is already declared as an entity. Furthermore, they use an *infol* tag to specify the type of BioCRelation being modelled, to explicitly distinguish event, relation, equivalence, and event modification. We have also adopted this representational choice. Finally, we have included an *infol* tag to specify the file extension of the annotation file (e.g. a1, a2 or ann). In the Peng et al proposal, the annotation type is instead used to identify the annotation file extension required to convert the BioC annotation back into the brat format. This dependency might be problematic if several file extensions are used in the future to define different sets of annotations for a given document.

Some questions remain about the best way to model document content in BioC arising from the difference identified in the application of the *passage* tag. The choice of the granularity of a “passage” in a document would seem to vary depending on what kind of text is annotated. While having a passage for each newline in the input may be appropriate for a short document such as an abstract and where newlines are consistently used to separate paragraphs, for some documents a different level of granularity could be more appropriate. For instance, a *passage* could more appropriately be an entire section/subsection within a document, or a set of paragraphs defined some other way. If the input text does not use newlines consistently to separate paragraphs (such as in the case for a LaTeX document, which uses two newlines rather than one to separate paragraphs), a *passage* might appropriately correspond to multiple input lines. A possible

---

<sup>5</sup>iSimp tool: <http://research.bioinformatics.udel.edu/isimp/>

solution for this would be to allow some specification of the appropriate definition of *passage* for a given conversion via a configuration parameter. This is left for future work.

### Entity annotation

Entity annotation in brat is mapped to the BioCAnnotation entity in BioC as shown in Figure 2. The type of the entity is provided with an *infony* tag with key value type and value the type of the entity. Start and end of the entity is mapped to offset and length in the BioC format. Support is provided for split entities by using several location entries in BioC.

```
brat
T1    disease 54 68 Lynch syndrome
BioC
<annotation id="T1">
<infony key="type">disease</infony>
<infony key="file">ann</infony>
<location offset="54" length="14"></location>
<text>Lynch syndrome</text>
</annotation>
```

**Figure 2.** Example of entity annotation in brat and BioC

### Relation annotation

A brat relation is encoded as a BioCRelation object in BioC as shown in Figure 3, and the brat id is used to identify the relation. Brat relations are binary, so only two nodes are created. The relation type is encoded as an *infony* object with key value *relation type* and the tag value contains the type of relation denoted in the brat format. Each related entity is encoded using the *node* tag, indicating the identifier of the entity in the *refid* attribute and the type of entity in the *role* attribute.

```
brat
R1_1  relatedTo body-part:T14 disease:15
BioC
<relation id="R1_1">
<infony key="type">relation</infony>
<infony key="relation type">relatedTo</infony>
<infony key="file">ann</infony>
<node refid="T14" role="body-part"></node>
<node refid="T15" role="disease"></node>
</relation>
```

**Figure 3.** Example of relation annotation in brat and BioC

### Event annotation

Events contain a relation between a trigger entity and one or more entities. This annotation type has been encoded using the `BioCRelation` as shown in Figure 4. The trigger and its type are encoded in a `node` tag while the related entities have been modelled as nodes as well. The relation id denotes the event identifier in the original file.

```
brat
E21  Negative_regulation:T48 Theme:E23
BioC
<relation id="E21">
<infon key="type">event</infon>
<infon key="file">a2</infon>
<infon key="event type">Negative_regulation</infon>
<node role="trigger" refid="T48"/>
<node role="Theme" refid="E23"/>
</relation>
```

**Figure 4.** Example of event annotation in brat and BioC

### Equivalence annotation

The equivalence entity relates to several entities, expressing that they are semantically equivalent. A `BioCRelation` is used to model the equivalence, mapping the related entities to node tags, without specific role. The `id` is set to `Equiv`. This is shown below in Figure 5.

```
brat
*    Equiv T6 T7
BioC
<relation id="Equiv">
<infon key="file">a2</infon>
<infon key="type">equiv</infon>
<node role="" refid="T6"/>
<node role="" refid="T7"/>
</relation>
```

**Figure 5.** Example of entity annotation in brat and BioC

### Attribute and modification annotation

This annotation type defines an attribute of another brat annotation. The same specification can work on several annotations. We have defined it as a `BioCRelation` and specified the type of the annotation using the attribute `type` in an `infon` tag. An example is shown below in Figure 6.

```

brat
M2   Negation E14
<relation id="M2">
<infony key="file">a2</infony>
<infony key="type">Negation</infony>
<node role="" refid="E14"/>
</relation>

```

**Figure 6.** Example of attribute and modification annotation in brat and BioC

### Normalization annotations

In addition to the boundaries of the entities, brat allows linking an identifier from a given resource to the annotated entities. The information provided as the annotation id, type of the annotation, the reference to the resource (in the example, Wikipedia) and a string linked to it are modelled using tags and attributes from the BioCRelation object. An example is shown below in Figure 7.

```

brat
N1   Reference T1 Wikipedia:534366   Barack Obama
BioC
<relation id="N1">
<infony key="file">a2</infony>
<infony key="string">Barack Obama</infony>
<infony key="type">Reference</infony>
<node role="Wikipedia:534366" refid="T1"/>
</relation>

```

**Figure 7.** Example of entity annotation in brat and BioC

### Note annotations

Brat allows adding annotations on the entities. The type and string are encoded as *infony* tags. The annotation on which the note is added is specified in a *node* tag. An example is shown below in Figure 8.

## Results

We have applied the conversion tool to existing corpora available in the brat format. The Brat2BioC tool is available from [https://bitbucket.org/nicta\\_biomed/brat2bioc](https://bitbucket.org/nicta_biomed/brat2bioc). The processed corpora include the HVP corpus [5], the BioNLP Shared Task 2009, 2011 and 2013, available from [https://bitbucket.org/nicta\\_biomed/brat2bioc/downloads](https://bitbucket.org/nicta_biomed/brat2bioc/downloads). The developed solution has been compared to the code available from the BioC website performing the transformation of the 2009

shared task data. Our software covers a larger set of brat annotations, thus it can deal with a large set of corpora.

```
brat
#1    AnnotatorNotes T1    this annotation is suspect
BioC
<relation id="#1">
<infony="file">a2</infony>
<infony="string">this annotation is suspect</infony>
<infony="type">AnnotatorNotes</infony>
<node role="" refid="T1"/>
</relation>
```

**Figure 8.** Example of note annotation in brat and BioC

In addition, Brat2BioC has been used to convert a large set of corpora which are available for visualization on the WBI repository<sup>6</sup>. This repository allows on-line visualization of more than 20 popular corpora on the biomedical natural language processing domain and annotations range from named-entities (e.g., genes and drugs) and binary relationships (e.g., protein-protein interactions) to biomedical events (e.g., phosphorylation). Most of these were converted to the BioC format and made available for download from repository's page, including the AIMed, BioInfer, BioText, CellFinder, Drug-Drug Interaction Extraction 2011, Drug-Drug Interaction Extraction 2013, GeneReg, Genia, GETM, GREC, HPDR50, IEPA, LLL, OSIRIS and SNP Corpus corpora. We have not converted those corpora whose license does not allow their redistribution and or those which are only available for download after license agreement (e.g., the SCAI chemical compound corpus).

## Conclusions

We have developed a tool to perform the conversion of the brat format into BioC. This conversion required analysing the way the information can be modelled in each system and explored the limitations of each of the annotation formalisms. Some possible configuration parameters, such as the extension for the generated annotation file, and a specification of the appropriate definition of a passage for the corpus, have been identified.

## Acknowledgements

This work was supported by Australian Federal and Victoria State Governments and the Australian Research Council through the ICT Centre of Excellence program, National ICT Australia (NICTA).

---

<sup>6</sup>WBI repository: <http://corpora.informatik.hu-berlin.de>

## References

1. Comeau, D et al (2013 to appear) "BioC: A Minimalist Approach to Interoperability for Biomedical Text Processing." Database: The Journal of Biological Databases and Curation.
2. Kim, Jin-Dong, et al. (2009) "Overview of BioNLP'09 shared task on event extraction." In Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing: Shared Task. Association for Computational Linguistics.
3. Neves, M., Damaschun, A., Kurtz, A., & Leser, U. (2012). "Annotating and evaluating text for stem cell research." In *Third Workshop on Building and Evaluation Resources for Biomedical Text Mining*.
4. Peng Y, Tudor C, Torii M , Wu CH, Vijay-Shanker K. (2013) "Enhance Interoperability of iSimp by Using the BioC Format." Submitted to the BioCreative IV workshop.
5. Verspoor K, et al. (2013) "Annotating the biomedical literature for the human variome." Database: the journal of biological databases and curation, bat019, doi:10.1093/database/bat019.

# A Biomedical Semantic Role Labeling BioC Module for BioCreative IV

Po-Ting, Lai<sup>1,2</sup>, Hong-Jie Dai<sup>3,\*</sup>, Johnny Chi-Yang Wu<sup>3</sup> and Richard Tzong-Han Tsai<sup>4,\*</sup>

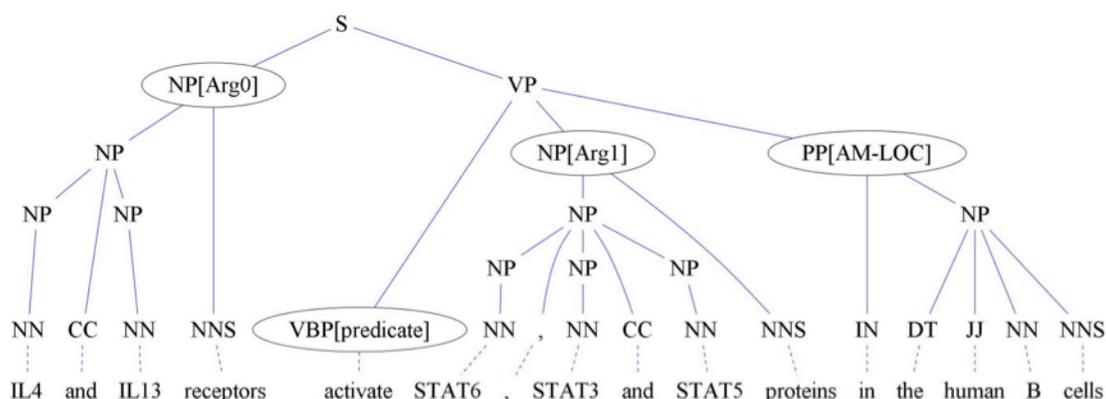
<sup>1</sup>Department of Computer Science, National Tsing-Hua University, HsinChu, Taiwan, R.O.C.,  
<sup>2</sup>Institute of Information Science, Academia Sinica, Taipei, Taiwan, R.O.C., <sup>3</sup>Graduate Institute of BioMedical Informatics, College of Medical Science and Technology, Taipei Medical University, Taipei, Taiwan, R.O.C., <sup>4</sup>Department of Computer Science and Information Engineering, National Central University, Taoyuan, Taiwan, R.O.C.

\*Corresponding author: E-mail: [hjdai@tmu.edu.tw](mailto:hjdai@tmu.edu.tw), [thtsai@csie.ncu.edu.tw](mailto:thtsai@csie.ncu.edu.tw)

## Abstract

With respect to the text mining community, the organizers of BioCreative IV have initiated the BioC project in attempt to propel researches within this area by providing a universal format for text mining tools. In this fashion, various tools performing distinct tasks can be integrated seamlessly in a less time- and effort-consuming manner. As a participant, we develop a semantic role labeling BioC module, which provides semantic analysis of biomedical literatures, hoping to benefit researchers with similar interest within the text mining field. The service is available at [http://bws.iis.sinica.edu.tw/BioC\\_BIOSMILE/BioC\\_Module.svc/SRL](http://bws.iis.sinica.edu.tw/BioC_BIOSMILE/BioC_Module.svc/SRL).

## Introduction



**Figure 1:** An example of the predicate-argument structure for the sentence “IL4 and IL13 receptors activate STAT6, STAT3, and STAT5 proteins in the human B cells”

Semantic role labeling (SRL) is a considerable technique in natural language processing, especially for life scientists who are interested in uncovering information related to biological processes within literatures. SRL represents a sentence by one or more predicate argument structures (PAS) [1]. Each PAS is composed of a predicate (e.g., a verb) and several arguments (e.g., noun phrases) that possess different semantic roles, including main arguments such as an agent<sup>1</sup> and a patient<sup>2</sup>, as well as adjunct arguments such as time, manner, and location. For example, the sentence in Figure 1 “IL4 and IL13 receptors activate STAT6, STAT3, and STAT5 proteins in the human B cells” describes a molecular activation process. It can be represented by a PAS in which “activate” is the predicate, the noun phrase “IL4 and IL13 receptors” constitutes the agent, “STAT6, STAT3, and STAT5 proteins” acts as the patient, and “in the human B cells” indicates the location of occurrence. Thus, the agent, patient, and location are all arguments of the predicate. SRL not only identifies the subjects involved in these processes, but also confirms the direction of existing interactions, along with supplementary manner, location or time details. Such knowledge is essential in comprehending signaling pathways behind versatile biological mechanisms and phenomena.

To make a contribution to the BioC repository of the BioCreative IV BioC track, we developed a SRL BioC module for biomedical literatures. The BioC module is an augmentation of our previous SRL system developed under the BioProp standard and corpus [2]. We also used it in

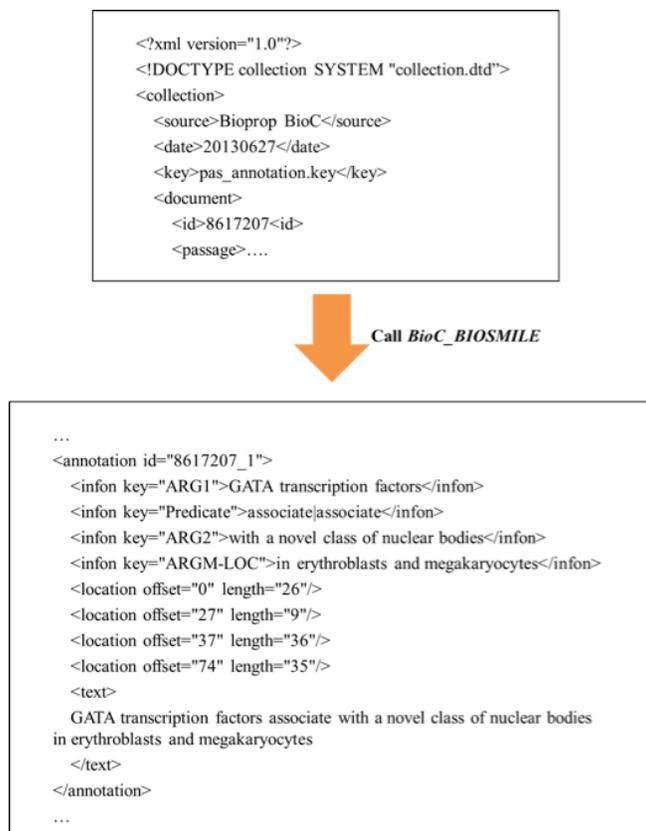
**Table 1:** The predicate and argument types are used

<b>Predicate</b>	
abolish, abrogate, accompany, act, activate, affect, alter, associate, augment, begin, bind, block, carry, catalyse, cause, clone, confer, conserve, contain, control, culture, decrease, delete, depend, derive, develop, differentiate, disrupt, down-regulate, eliminate, encode, enhance, exert, express, function, generate, include, increase, induce, influence, inhibit, initiate, interact, interfere, involve, isolate, lack, lead, link, lose, mediate, modify, modulate, mutate, participate, phosphorylate, play, prevent, produce, proliferate, promote, purify, recognize, reduce, regulate, repress, require, result, reveal, signal, skip, splice, stimulate, suppress, target, transactivates, transcribe, transfect, transform, trigger, truncate, up-regulate,	
<b>Argument Type</b>	
Arg0	agent
Arg1	direct object/theme/patient
Arg2-5	not fixed
ArgM-NEG	negation marker
ArgM-LOC	location
ArgM-TMP	time
ArgM-MNR	manner
ArgM-EXT	extent
ArgM-ADV	general-purpose
ArgM-PNC	purpose
ArgM-CAU	cause
ArgM-DIR	direction
ArgM-DIS	discourse connectives
ArgM-MOD	modal verb
ArgM-REC	reflexives and reciprocals
ArgM-PRD	marks of secondary predication

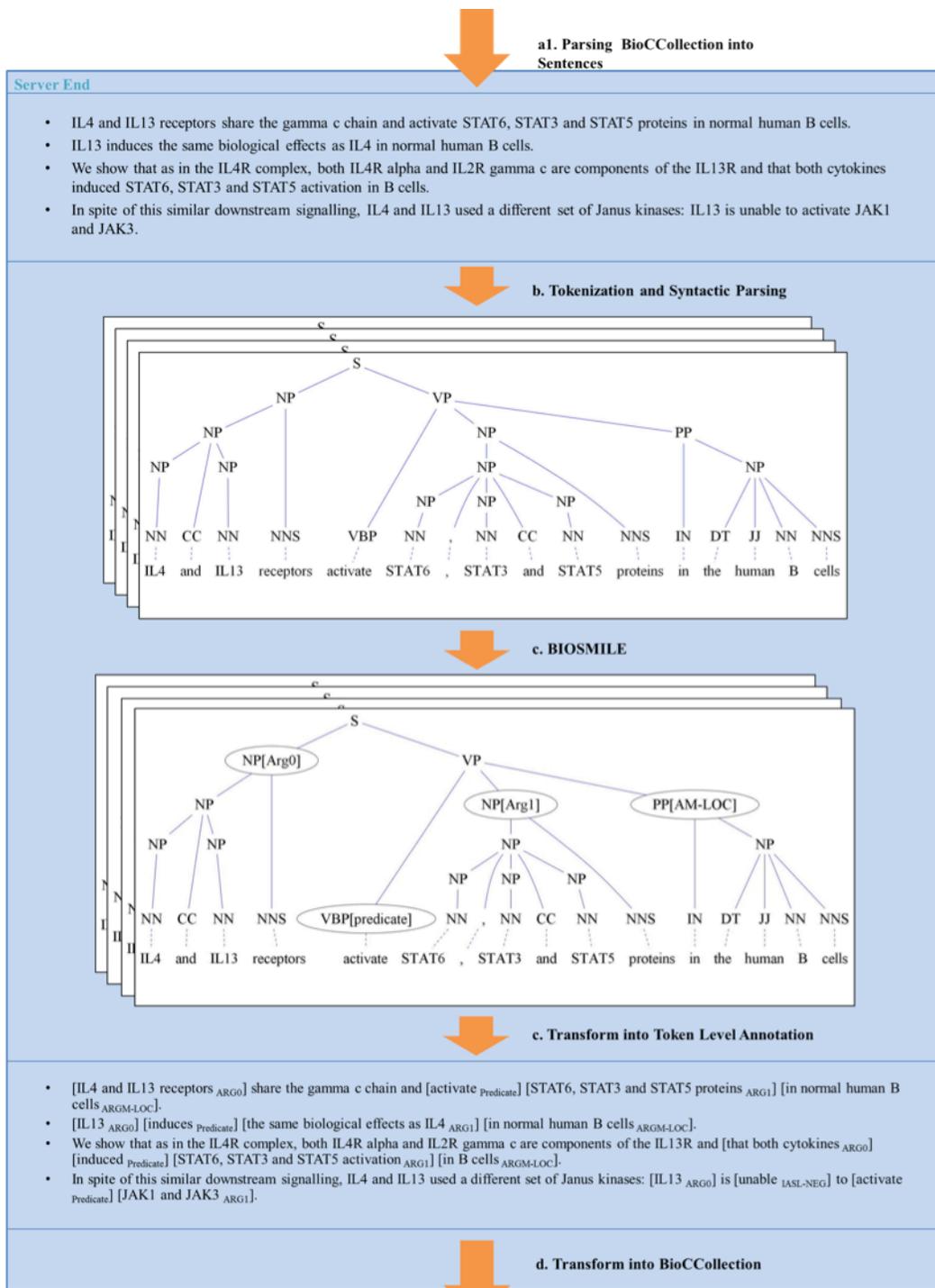
our previous web services, including BIOSMILE Web Search [3], PubMed-EX [4] and T-HOD [5]. The module supports 82 predicates and 32 argument types, with the latter manually defined as location, manner, temporal etc. Please refer to Table 1 for further details. An online demo system and additional information of the developed module will be available at [http://bws.iis.sinica.edu.tw/BioC\\_BIOSMILE/BioC\\_Module.svc/SRL](http://bws.iis.sinica.edu.tw/BioC_BIOSMILE/BioC_Module.svc/SRL).

## Semantic Role Labeling BioC Module

Our SRL BioC module allows clients to submit one or more articles online, and the server will return the SRL results in BioC format. Clients only need to provide the article information in the BioC format (an example is shown in Figure 2). Tokenization and syntactic structure information will be generated on the server, and our BIOMedical SeManTic LabElling (BIOSMILE) system will produce the SRL results accordingly. Further interpretation of the results will not be necessary, since the SRL annotation is displayed independently with the syntactic structure by using the offset information in the BioC format. Subsequently, we will use Figure 3 as an example to demonstrate how the module process BioC articles. Afterwards, we will refer to our module as “BioC\_BIOSMILE”.



**Figure 2:** BioC\_BIOSMILE uses offsets to indicate the position of arguments/predicate. For instance, the phrase “with a novel class of nuclear bodies” is ARG2 with its start index at 37 (3<sup>rd</sup> node) and a length of 36 characters. Infons are shown according to the sequential order of arguments/predicate in the sentence.



**Figure 3:** The procedure of BioC\_BIOSMILE processing.

In Figure 3, after calling BioC\_BIOSMILE, the server will first process the articles in the article collection encoded in the BioC format. Next, we will use the LingPipe sentence splitter toolkit to determine the boundaries of sentences. Following sentence splitting, we apply the GENIA Tagger and a biomedical full parser based on the Charniak parser to construct the syntactic

structure. Based on the information above, our BIOSMILE system then generates semantic roles for each constituent on the parse tree. Lastly, we transform the constituent annotation results into token level results, and return them to the client sides in the BioC format.

## Usage Scenario

Our module can support many biomedical natural language processing groups to develop or improve their systems. Here we use two tasks, protein-protein interaction (PPI) extraction and biomedical event extraction, to demonstrate how our tool can benefit these researches.

### Protein-protein Interaction Extraction

PPI extraction is a classical binary classification problem[6], which determines the relation type between two proteins as POSITIVE or NEGATIVE. From the perspective of SRL, the relation entity pairs usually possess an agent1 and patient2 relation, with the relation keyword as the predicate. To develop a system performing this task, the machine-learning based approach can easily utilize the argument-combinations as features. Take a PPI related sentence “IL4 and IL13 receptors activate STAT6, STAT3 and STAT5 proteins in the mouse cells.” as an example. The interpretation of the SRL module is as follows.

```
<annotation>
  <infol key="ARG0">IL4 and IL13 receptors</infol>
  <infol key="Predicate">activate</infol>
  <infol key="ARG1">STAT6, STAT3 and STAT5 proteins</infol>
  <infol key="ARGM-LOC">in the mouse cells</infol>
  <location offset="0" length="22" />
  <location offset="23" length="8" />
  <location offset="32" length="31" />
  <location offset="64" length="18" />
  <text>IL4 and IL13 receptors activate STAT6, STAT3 and STAT5 proteins in the mouse
  cells</text>
</annotation>
```

It can be observed that the objects of the PPI pair are in (ARG0, ARG1) and associated with a predicate “activate”.

### Biomedical Event Extraction

The goal of biomedical event extraction[7] is to retrieve knowledge regarding biomedical activities from literatures. For instance, the interpretation of the sentence[8] “BMP-6 rapidly induced phosphorylation of Smad1/5/8” is shown in Figure 4.

**Figure 4:** The interpretation of “BMP-6 rapidly induced phosphorylation of Smad1/5/8” visualized with Brat.

From the perspective of SRL, the module will interpret it as:

```

<annotation>
  <infol key="ARG0">BMP-6</infol>
  <infol key="ARGM-MNR">rapidly</infol>
  <infol key="Predicate">induced</infol>
  <infol key="ARG1">phosphorylation of Smad1/5/8</infol>
  <location offset="0" length="5" />
  <location offset="6" length="7" />
  <location offset="14" length="7" />
  <location offset="22" length="28" />
  <text>BMP-6 rapidly induced phosphorylation of Smad1/5/8</text>
</annotation>

```

## Technical Details

### Semantic Role Labeling System

Our system formulates the SRL task as a constituent-by-constituent labeling problem, and uses a machine-learning-based classifier to assign a proper semantic role for each constituent (those without semantic roles will be assigned NULL). Our SRL system achieved an F-score of 84.76%

on the BioProp corpus. The evaluation metrics is defined as  $F = \frac{2 \times P \times R}{P + R}$ , where  $P$  denotes the precision and  $R$  denotes the recall. The formulae for calculating precision and recall are as follows:

$$\text{Precision} = \frac{\text{the number of correctly recognized arguments}}{\text{the number of recognized arguments}}$$

$$\text{Recall} = \frac{\text{the number of correctly recognized arguments}}{\text{the number of true arguments}}$$

### A Test Dataset for BioC

In addition to our previous evaluation, we provide a small test dataset consisting of 50 abstracts in the BioC format. These abstracts are randomly selected from GTB, and domain experts were employed to manually annotate the semantic role labels of 52 predicates. Further statistical details of our dataset are shown in Table 2, and the performance is shown in Table 3.

**Table 2:** The statistics of SRL BioC corpus

<b>Role</b>	<b>Number</b>
Core argument types	11
Adjunctive argument types	21
<b>Other</b>	<b>Number</b>
Event types	52
Abstracts with Propositions	50
Propositions	254

**Table 3:** The performance of BIOSMILE on SRL BioC corpus

<b>#True positive</b>	445
<b>#False positive</b>	149
<b>#False negative</b>	192
<b>Precision</b>	75%
<b>Recall</b>	70%
<b>F-score</b>	72%

## Acknowledgements

This work was supported by the National Science Council of Taiwan under the grant number NSC102-2319-B-010-002, NSC-102-2218-E-038-001, and the Taipei Medical University under the grant number TMU101-AE1-B55.

## References

1. P. Kingsbury and M. Palmer (2002) From Treebank to PropBank. *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002)*, pp. 1989–1993.
2. W.-C. Chou, R. T.-H. Tsai, Y.-S. Su, W. Ku, T.-Y. Sung, and W.-L. Hsu (2006) A Semi-Automatic Method for Annotating a Biomedical Proposition Bank. the *Proceedings of ACL Workshop on Frontiers in Linguistically Annotated Corpora*, Sydney, Australia
3. H.-J. Dai, C.-H. Huang, R. T. K. Lin, R. T.-H. Tsai, and W.-L. Hsu (2008) BIOSMILE web search: a web application for annotating biomedical entities and relations. *Nucl. Acids Res.*, vol. 36, pp. W390-W398.
4. R. T.-H. Tsai, H.-J. Dai, P.-T. Lai, and C.-H. Huang (2009) PubMed-EX: A web browser extension to enhance PubMed search with text mining features. *Bioinformatics*, vol. 25, pp. 3031-3032.
5. H.-J. Dai, C.-Y. Wu, R. T.-H. Tsai, W.-H. Pan, and W.-L. Hsu (2013) T-HOD: A Literature-based Candidate Gene Database for Hypertension, Obesity, and Diabetes. *Database: The Journal of Biological Databases and Curation*.
6. M. Miwa, R. Sætre, Y. Miyao, and J. i. Tsujii (2009) Protein–protein interaction extraction by leveraging multiple kernels and parsers. *International Journal of Medical Informatics*, vol. 78, pp. e39-e46.
7. J.-D. Kim, S. Pyysalo, T. Ohta, R. Bossy, N. Nguyen, and J. i. Tsujii (2011) Overview of BioNLP Shared Task 2011. the *Proceedings of the BioNLP Shared Task 2011 Workshop*, Portland, Oregon..
8. P. Stenetorp, S. Pyysalo, G. Topić, T. Ohta, S. Ananiadou, and J. i. Tsujii (2012) BRAT: a web-based tool for NLP-assisted text annotation. the *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, Avignon, France.

# NaCTeM's BioC Modules and Resources for BioCreative IV

Rafal Rak\*, Riza Batista-Navarro, Andrew Rowley, Makoto Miwa, Jacob Carter and Sophia Ananiadou

National Centre for Text Mining, University of Manchester, UK

\*Corresponding author: Tel: +441613063090, E-mail: rafal.rak@manchester.ac.uk

## Abstract

Participating in the Interoperability track of BioCreative IV, we have chosen to focus on three biomedical natural language processing (NLP) tasks: 1) Extraction of biomolecular events, 2) Identification of metabolic process concepts and 3) Recognition of concepts in the Comparative Toxicogenomics Database (CTD). Our contribution towards the first task consists of a module for automatically extracting events. For the second, we prepared a module based on components used in our participation in the User Interactive track. RESTful web services recognising chemicals, genes, diseases and action terms contribute towards the last task. Together with these BioC-compliant modules, several corpora in the BioC format have been made available.

**Keywords:** Interoperability, BioC, Workflows, Web services, Event extraction, Concept identification, CTD concept recognition, BioNLP Shared Tasks, Metabolic processes

## Introduction

The BioCreative IV Interoperability Initiative aims at enhancing the reusability of tools and resources by promoting a common data interchange format, BioC. The format is encoded in XML and consists of a collection of documents, each split into passages and optionally sentences. These elements may contain stand-off annotations with optional text-bound locations as well as *n*-ary relations between annotations and other relations. Virtually all elements may declare *infos*, a list of key-value pairs.

In order to foster this initiative, we transcribed several resources relevant to biology and biochemistry, most of which were originally prepared by the National Centre for Text Mining (NaCTeM). They are selected BioNLP Shared Task 2011- and 2013-edition<sup>1</sup> corpora and the Metabolites corpus (1). The BioNLP Shared Task corpora come from the Infectious Diseases (ID) (2) and Epigenetics and Post-translational Modifications (EPI) (3) tasks ran in the 2011 edition, the Cancer Genetics (CG) (4) and Pathway Curation (PC) (5) tasks ran in the 2013 edition, as well as the GENIA tasks (GE'11 (6) and GE'13 (7)) ran in both editions.

---

<sup>1</sup> <https://sites.google.com/site/bionlpst>, <http://2013.bionlp-st.org>

We created modules capable of automatically extracting information relevant to the aforementioned resources, as well as a series of web services that was developed for the BioCreative's CTD track<sup>2</sup>.

## Methods

All of the resources were produced in Argo<sup>3</sup>, a web-based, text mining workbench (8). Argo allows users to build their task-specific processing workflows from a library of elementary analytics (e.g., data readers/writers, syntactic and semantic analytics). The platform is based on the Unstructured Information Management Architecture (UIMA) (9) which ensures the interoperability of elementary analytics (processing components) by imposing common type systems (annotation schemata).

Similarly, most of the proposed modules are available in Argo as workflows, i.e., arrangements of processing components that form meaningful, self-contained processing units. To facilitate the support for BioC format, we developed the BioC type system, which encodes this format in UIMA, as well as two processing components, BioC Reader and BioC Writer, that are capable of (de)serialising BioC collections from/to the type system. The two BioC components utilise an API provided by the creators of the format<sup>4</sup>.

## Resources

*BioNLP Shared Task corpora.* The main subjects of annotations included in the BioNLP Shared Task corpora are biologically relevant named entities (e.g., proteins, organs, DNA), and named biological processes, i.e., events, which associate a trigger word or phrase signalling a process (e.g., *activation*, *inhibits*) with named entities participating in the process. Each of the event participants is labelled with the role it plays in the event (e.g., theme, cause). Events may also be enriched with attributes that modify their interpretation, namely, speculation and negation. Additional annotations include the equivalence of entities (usually associations between abbreviations and their expanded forms that appear in close proximity in text) and, in the case of GE<sup>13</sup>, coreferences encoded as binary relations between an anaphoric expression and its antecedent.

*Metabolites corpus.* We also make available as a resource NaCTeM's corpus of 296 MEDLINE abstracts enriched with entity annotations corresponding to metabolites and enzymes (9). Previously used in a pilot study on yeast metabolic network reconstruction (10), the documents were manually annotated by two domain experts who were asked to mark up names of enzymes and metabolites only if they appear in the context of metabolic pathways. Originally released in

---

<sup>2</sup> <http://www.biocreative.org/tasks/biocreative-iv/track-3-CTD>

<sup>3</sup> <http://argo.nactem.ac.uk>

<sup>4</sup> <http://bioc.sourceforge.net>

the MEDLINE XML format<sup>5</sup>, the corpus was first converted to an intermediate format, i.e., that of the BioNLP Shared Task, taking only metabolite entity annotations.

*BioNLP to BioC conversion.* In order to read the data in the BioNLP Shared Task format, which is encoded in plain-text files, we used the BioNLP Shared Task Data Reader component available in Argo, built specifically for the shared tasks (11). The reader encodes data into the tailored BioNLP type system. In order to transcribe data between the BioNLP type system and that of BioC, we used the SPARQL Annotation Editor component that allows a workflow designer to manipulate annotations represented as an RDF graph (12).

The BioNLP annotations were encoded using the annotation and relation elements available in BioC. Both annotations and relations include information about their type, which can be one of “Entity” and “Trigger” for annotations, and “Event”, “Equivalent” and “Coreference” for relations. Table 1 shows the example snippets of BioC syntax for each of the BioNLP annotations.

**Table 1.** Examples of the transcription of BioNLP annotations into BioC XML format.

BioNLP annotations	BioC transcription
Entities	<pre>&lt;annotation id="T1"&gt; &lt;infony="type"&gt;Entity&lt;/infony&gt; &lt;infony="category"&gt;Protein&lt;/infony&gt; &lt;location offset="19" length="30"/&gt; &lt;text&gt;interferon regulatory factor 4&lt;/text&gt; &lt;/annotation&gt;</pre>
Event triggers	<pre>&lt;annotation id="TRIGGER_55_65"&gt; &lt;infony="type"&gt;Trigger&lt;/infony&gt; &lt;location offset="55" length="10"/&gt; &lt;text&gt;expression&lt;/text&gt; &lt;/annotation&gt;</pre>
Events and event modifications	<pre>&lt;relation id="E2"&gt; &lt;infony="type"&gt;Event&lt;/infony&gt; &lt;infony="category"&gt;Gene_expression&lt;/infony&gt; &lt;infony="negation"&gt;&gt;false&lt;/infony&gt; &lt;infony="speculation"&gt;&gt;false&lt;/infony&gt; &lt;node refid="TRIGGER_55_65" role="EventTrigger"/&gt; &lt;node refid="T1" role="Theme"/&gt; &lt;/relation&gt;</pre>

<sup>5</sup> [http://www.nlm.nih.gov/bsd/licensee/data\\_elements\\_doc.html](http://www.nlm.nih.gov/bsd/licensee/data_elements_doc.html)

Equivalent entities	<pre> &lt;relation id="EE53"&gt; &lt;infony key="type"&gt;Equivalent&lt;/infony&gt; &lt;node refid="T2" role=""/&gt; &lt;node refid="T3" role=""/&gt; &lt;/relation&gt; </pre>
Coreferences (GENIA corpora)	<pre> &lt;relation id="RT13"&gt; &lt;infony key="type"&gt;Coreference&lt;/infony&gt; &lt;node refid="T13" role="Subject"/&gt; &lt;node refid="T3" role="Object"/&gt; &lt;node refid="T4" role="Object"/&gt; &lt;node refid="T5" role="Object"/&gt; &lt;/relation&gt; </pre>

Special consideration was given to coreferences; instead of encoding them as binary relations (as is the case in the original data), we combined all entities that appear as “objects” (antecedents) of the same “subject” (anaphoric expression) into a single BioC relation, taking full advantage of its *n*-ary representation.

In addition to these annotations, each generated BioC file also contains collection-level data such as source information, date, key location, versions, and licence.

## Modules

We prepared two workflows in Argo for event extraction and metabolic process concept identification. Each workflow includes the BioC Reader and Writer components that allow users to upload their BioC files for processing as well as harvest the results in the same format.

*Event extraction.* The BioNLP Shared Task event extraction workflow involves the Enju parser (13), GENIA dependency parser (14), and EventMine. EventMine is an adaptable, machine learning-based event extraction system that achieved the best performance on the GE’11, EPI, ID and PC data sets and the second best on the CG data set (15). It performs a series of classifications for event trigger recognition, argument identification and role assignment. Additionally, it is capable of resolving coreferences (16) and recognising event modifications such as negation and speculation (17).

The workflow assumes that the BioC collection given as input already contains annotated bioentities, as defined in the BioNLP Shared Task series. The output of the workflow consists of documents enriched with information pertaining to biomolecular events (i.e., event triggers, participants, modifications) and relations (e.g., equivalent entities). The EventMine component allows the user to choose a model tailored for a specific extraction task (one of GE, EPI, ID, PC, CG), which ultimately defines the output event types.

*Metabolic process concept identification.* Building upon the workflow we have prepared for our participation in BioCreative's User Interactive track<sup>6</sup>, we adapted NLP components capable of enriching documents with annotations relevant to metabolic processes. These include a refactored version of OSCAR 3 (18) for recognising metabolites and metabolic process expressions (e.g., *hydroxylates*, *deacetylated*) and GENIA Tagger (19) for recognising genes and gene products (GGPs). The recognised concepts are then linked to unique identifiers in external databases using components based on the Jaro-Winkler string similarity algorithm. Metabolites are linked to entries in ChEBI (20), GGPs to UniProt (21), and metabolic process expressions to the CTD interaction types ontology<sup>7</sup>.

The workflow for this task requires only text in the BioC format. The output annotations consist of the recognised concepts, each of which is tagged with the corresponding identifier in the relevant external database.

*CTD concept recognition.* Following the specifications of the BioCreative's CTD track, we developed RESTful, BioC-compliant web services which recognise the following concepts in the CTD (22): chemicals, genes, diseases and action terms. Our named entity recognisers for the first three types are based on the conditional random fields (CRFs) algorithm (23) and implemented on top of the NERsuite package<sup>8</sup>. In recognising action terms, in contrast, we took a multiclass, multilabel approach based on the support vector machines (SVMs) algorithm (24). The features used are described in detail in our report for the CTD track (25).

Each of the web services accepts a request in the BioC format and sends back a BioC XML response containing annotations of one of the four concept types.

## Availability

The modules and resources can be accessed by following the instructions in the Argo BioC web page<sup>9</sup>.

## Funding

This work was partially supported by Europe PubMed Central funders (led by Wellcome Trust).

---

<sup>6</sup> <http://www.biocreative.org/tasks/biocreative-iv/track-5-IAT>

<sup>7</sup> <http://ctdbase.org/help/ixnQueryHelp.jsp#actionType>

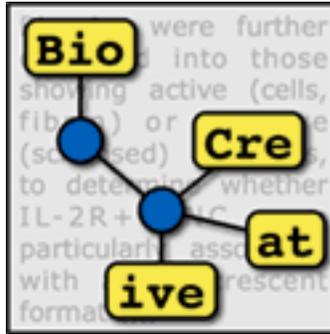
<sup>8</sup> <http://nersuite.nlplab.org>

<sup>9</sup> <http://argo.nactem.ac.uk/bioc>

## References

1. Nobata, C., et al., *Mining Metabolites: Extracting the Yeast Metabolome from the Literature*. *Metabolomics*, 2011. **7**(1): p. 94-101.
2. Pyysalo, S., et al. *Overview of the Infectious Diseases (ID) task of BioNLP Shared Task 2011*. In *Proceedings: BioNLP Shared Task 2011 Workshop*. 2011. Association for Computational Linguistics.
3. Ohta, T., S. Pyysalo, and J. Tsujii. *Overview of the Epigenetics and Post-translational Modifications (EPI) task of BioNLP Shared Task 2011*. In *Proceedings: BioNLP Shared Task 2011 Workshop*. 2011. Association for Computational Linguistics.
4. Pyysalo, S., T. Ohta, and S. Ananiadou. *Overview of the Cancer Genetics (CG) task of BioNLP Shared Task 2013*. In *Proceedings: BioNLP Shared Task 2013 Workshop*. 2013. Sofia, Bulgaria: Association for Computational Linguistics.
5. Ohta, T., et al. *Overview of the Pathway Curation (PC) task of BioNLP Shared Task 2013*. In *Proceedings: BioNLP Shared Task 2013 Workshop*. 2013. Sofia, Bulgaria: Association for Computational Linguistics.
6. Kim, J.-D., et al., *Overview of Genia event task in BioNLP Shared Task 2011*, in *Proceedings of the BioNLP Shared Task 2011 Workshop 2011*, Association for Computational Linguistics: Portland, Oregon. p. 7-15.
7. Kim, J.D., Y. Wang, and Y. Yasunori. *The Genia Event Extraction Shared Task, 2013 Edition - Overview*. In *Proceedings: BioNLP Shared Task 2013 Workshop*. 2013. Sofia, Bulgaria: Association for Computational Linguistics.
8. Rak, R., et al., *Argo: an integrative, interactive, text mining-based workbench supporting curation*. *Database*, 2012. **2012**.
9. Ferrucci, D. and A. Lally, *UIMA: an architectural approach to unstructured information processing in the corporate research environment*. *Nat. Lang. Eng.*, 2004. **10**(3-4): p. 327-348.
10. Markus, J.H., et al., *A consensus yeast metabolic network reconstruction obtained from a community approach to systems biology*. *Nature Biotechnology*, 2008. **26**(10): p. 1155-1160.
11. Nédellec, C., et al. *Overview of BioNLP Shared Task 2013*. In *Proceedings: BioNLP Shared Task 2013 Workshop*. 2013. Sofia, Bulgaria.
12. Rak, R. and S. Ananiadou. *Making UIMA Truly Interoperable with SPARQL*. In *Proceedings: 7th Linguistic Annotation Workshop and Interoperability with Discourse*. 2013. Sofia, Bulgaria.
13. Miyao, Y., et al. *Task-Oriented Evaluation of Syntactic Parsers and Their Representations*. In *Proceedings: ACL-08:HLT*. 2008.
14. Sagae, K. and J. Tsujii. *Dependency parsing and domain adaptation with LR models and parser ensembles*. In *Proceedings: CoNLL 2007 Shared Task in the Joint Conferences on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL'07 shared task)*.
15. Miwa, M. and S. Ananiadou. *NaCTeM EventMine for BioNLP 2013 CG and PC tasks*. In *Proceedings: BioNLP Shared Task 2013 Workshop*. 2013. Sofia, Bulgaria: Association for Computational Linguistics.
16. Miwa, M., P. Thompson, and S. Ananiadou, *Boosting automatic event extraction from the literature using domain adaptation and coreference resolution*. *Bioinformatics*, 2012. **28**(13): p. 1759-1765.

17. Miwa, M., et al., *Extracting semantically enriched events from biomedical literature*. BMC Bioinf., 2012. **13**: p. 108.
18. Kolluru, B., et al., *Using Workflows to Explore and Optimise Named Entity Recognition for Chemistry*. PLoS ONE, 2011. **6**(5): p. e20181.
19. Tsuruoka, Y., et al. *Developing a Robust Part-of-Speech Tagger for Biomedical Text*. In Proceedings: *Advances in Informatics - 10th Panhellenic Conference on Informatics*. 2005. Springer-Verlag.
20. Hastings, J., et al., *The ChEBI reference database and ontology for biologically relevant chemistry: enhancements for 2013*. Nucleic Acids Res., 2013. **41**(D1): p. D456-D463.
21. Consortium, T.U., *Update on activities at the Universal Protein Resource (UniProt) in 2013*. Nucleic Acids Res., 2013. **41**(D1): p. D43-D47.
22. Davis, A.P., et al., *The Comparative Toxicogenomics Database: update 2013*. Nucleic Acids Res., 2013. **41**(D1): p. D1104-D1114.
23. Lafferty, J.D., A. McCallum, and F.C.N. Pereira. *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data*. In Proceedings: *Eighteenth International Conference on Machine Learning*. 2001. Morgan Kaufmann Publishers Inc.
24. Joachims, T., *Learning to Classify Text Using Support Vector Machines. Methods, Theory and Algorithms*. Vol. 668. 2002: Springer.
25. Batista-Navarro, R.T., R. Rak, and S. Ananiadou. *NaCTeM CTD Web Services*. In Proceedings: *BioCreative IV Challenge and Workshop*. 2013. Bethesda, Maryland, USA.



## TRACK 3 (CTD)

### Organizers:

- Thomas Wieggers, North Carolina State University, USA
- Carolyn J. Mattingly, North Carolina State University, USA
- Allan P. Davis, North Carolina State University, USA

# Web services-based text mining demonstrates broad impacts for interoperability and process simplification

Thomas C. Wieggers<sup>\*</sup>, Allan Peter Davis, and Carolyn J. Mattingly

Department of Biology, North Carolina State University, Raleigh, NC 27695-7617, USA

<sup>\*</sup>Corresponding author: Tel: 207 288 9880, E-mail: tcwieger@ncsu.edu

## Abstract

The Critical Assessment of Information Extraction systems in Biology (BioCreative IV) challenge evaluation tasks collectively represent a community-wide effort for evaluating a wide variety of text mining and information extraction systems applied to the biological domain. The 'BioCreative IV Workshop' was comprised of five independent but largely complementary subject areas, including Track 3, which focused on Comparative Toxicogenomics Database (CTD)-related (<http://ctdbase.org>) named-entity recognition (NER). The CTD group organized document ranking and NER-related tasks for 'BioCreative Workshop 2012', and a key finding of the effort was that interoperability and integration complexity were major impediments to the direct application of the collaboration to the CTD text-mining pipeline and underscored a common problem with software development efforts. Major interoperability-related issues included lack of process modularity, operating system incompatibility, tool configuration complexity, and lack of standardization of high level inter-process communications. One way to potentially mitigate interoperability and general integration issues is the use of Web services to abstract implementation details; rather than integrating NER tools directly, make HTTP-based calls from CTD's asynchronous, batch-oriented text-mining pipeline to remote NER Web services for recognition of specific biological terms using BioC (an emerging family of simple XML formats) for inter-process communications. To test this concept, participating groups were asked to develop Representational State Transfer (REST)/BioC-compliant Web services tailored to CTD's NER requirements. Participants were provided with a comprehensive set of training materials. CTD staff evaluated the remote Web service-based URLs against a test dataset of 510 manually curated scientific articles. Twelve groups participated in the challenge. Recall, precision, balanced F-scores, and response times were calculated for each Web service. Top balanced F-scores for gene, chemical, and disease NER were 61%, 74%, and 51%, respectively. Response times ranged from fractions of a second to over 60 seconds per article. Here we present a detailed description of the challenge and summary of the results.

## Introduction

The Comparative Toxicogenomic Database (CTD) is a publicly available, manually curated resource that promotes understanding of the mechanisms by which drugs and environmental

chemicals influence biological processes and human health [1]. CTD's PhD-level staff biocurators review the scientific literature and manually curate chemical-gene/protein interactions, chemical-disease relationships, and gene-disease relationships, using a novel, highly structured notation in conjunction with CTD's Web-based curation tool [2]. The manual curation process organizes disparate data from scientific publications into a standard, structured format, making it more manageable and computable for bioinformatics-related processing. Curated data are integrated with other external datasets to facilitate development of novel hypotheses about chemical-gene-disease networks [1].

Curated data are captured using publicly available controlled vocabularies. Diseases are represented using CTD's disease vocabulary, MEDIC [3], that merges OMIM [4] terms with the *Disease* subset of the National Library of Medicine's Medical Subject Headings (MeSH) vocabulary [5]; genes/proteins are represented using Entrez Gene [6]; chemicals/drugs are represented using a modified subset of *Chemicals and Drugs* of MeSH [5]; and chemical-gene/protein interactions are captured using CTD's action term vocabulary [1].

CTD typically selects curation topics by targeting specific chemicals from a 'Chemical Priority Matrix'[2]. Depending on the chemical targeted, there are often many more articles available for curation than can be realistically curated by CTD biocurators. For example, a recent query for 'arsenic' at the PubMed interface from the National Center for Biotechnology Information (NCBI) Web site yielded over 20,000 scientific articles, many more documents than CTD could reasonably manually curate. In order to ensure that biocurators review only those articles that are most likely to yield curatable information within the context of CTD's structured curation paradigm, CTD staff assessed the feasibility and potential advantages of implementing a text-mining pipeline [7]. Based on the results of this study, CTD staff designed, developed, documented and implemented a highly effective, fully functional text-mining pipeline [8]. At the heart of the CTD text-mining pipeline is an internally developed, rules-based ranking algorithm that scores each article with a document relevancy score (DRS). Integral to the ranking algorithm is a set of third party NER tools adapted for CTD use: Abner [9] for gene NER; Oscar3 [10, 11] for chemical NER; and MetaMap [12] for disease recognition, as well as supplementary chemical and gene recognition. The effective deployment of these NER tools is essential to the success of DRS-based scoring in that the algorithm scores articles based in part by the pervasiveness and spatial orientation of CTD's controlled vocabulary terms in the text of the article's abstract.

CTD is constantly exploring new ways to improve the effectiveness of DRS scoring. The 'BioCreative Workshop 2012' Track I/Triage workshop focused on document triaging for CTD [13]. More specifically, participants developed tools that ranked articles in terms of their curatability, and identified gene/protein, chemical/drug, and disease actors, as well as CTD interaction-related action terms. Although the results were impressive, they were of little direct

benefit to CTD because NER tools developed by Track I participants were written using a wide variety of technologies and within technical infrastructures and architectures that would not necessarily easily integrate into CTD's existing text-mining pipeline. In short, interoperability and integration complexity were major impediments to the direct application of the NER-related aspects of the collaboration to the CTD pipeline. Impediments included lack of NER process modularity, operating system and programming language incompatibility, tool configuration complexity, lack of standardization of high level inter-process communications, and database management system-related incompatibility.

One alternative to potentially mitigate NER-related interoperability and general integration issues is the use of Web services, which are designed to accommodate interoperable machine-to-machine interaction over the Web [14]. More specifically, rather than integrating NER tools directly into the CTD text-mining pipeline, Web services provide the capability to make simple HTTP-based calls from CTD's asynchronous, batch-oriented text-mining pipeline to remote NER Web services for gene/protein, chemical/drug, disease, and chemical/gene-specific action term recognition. This approach tends to be inherently simpler than direct pipeline integration because the technical details of the tools themselves are completely abstracted by the Web service. Alternatively, direct integration requires text-mining pipeline developers to concern themselves with issues like operating system compatibility, programming language interpretation/compilation-related environments, tool versioning maintenance and control, tool-associated library compatibility, tool configuration maintenance, process modularity, inter-process communications, *etc.* The potential benefits of conceptual Web serviced-based NER capability for CTD and other Web-based resources were sufficient to use Track 3 as a mechanism to further study this approach.

As Track 3 tasks were being analyzed and designed by CTD staff, a group of NCBI-led collaborators were concurrently working on the development of a common interchange format to represent, store, and exchange data in a simple manner between different language processing systems and text-mining tools. This collaboration led to the development of BioC, a family of simple XML formats, to share text documents and annotations [15]. BioC's lightweight, flexible design, along with its support across multiple programming languages, made it a suitable vehicle for Track 3 inter-process communications.

The CTD track of BioCreative IV focused on NER interoperability and tool complexity abstraction. Participants were asked to build interoperable, Web service-based tools that would enable CTD to send text passages to their remote sites in order to identify gene/protein, chemical/drug, disease, and chemical/gene-specific action term mentions, each within the context of CTD's controlled vocabulary structure, using BioC for inter-process communications. The challenge was to determine whether teams such as CTD could benefit from text-mining tools developed using a common interoperable communications framework? If so, would the response

time associated with such tools be suitable for asynchronous, batch processing-based text-mining using technologies such as Web services?

## Methods and Materials

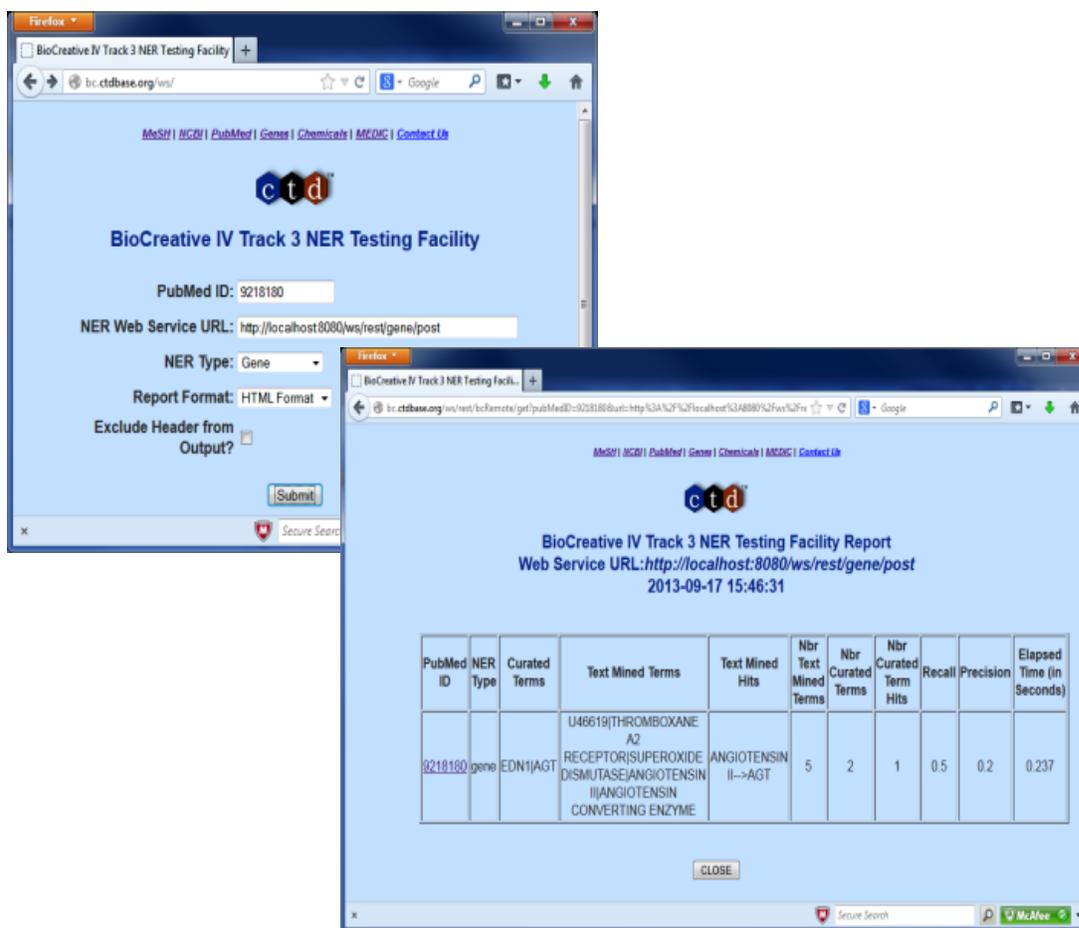
### *Web Service Architectural Style*

Representational State Transfer (REST) was selected as the architectural style for the participant Web services. REST was designed to abstract the architectural elements of distributed systems by enabling client processes to ignore the details of component implementation and protocol syntax to order to instead focus on the role of the components, enhancing simplicity by providing a clean separation of concerns, and hiding the underlying implementation of resources and communication mechanisms [16]. The primary purpose of REST-compliant Web services is to manipulate XML representations using a uniform set of stateless operations [17]; the term stateless in the Web service context means that requests are processed without knowledge of any prior requests. The stateless nature of REST tends to improve scalability because the Web service need not store state information between requests, allowing the server component to quickly free resources; moreover, the stateless feature simplifies implementation because the server need not manage resource usage across requests [16]. Although other Web service-based options were available, the timely emergence of BioC, coupled with REST's XML-centric nature and other attractive design features, made a REST/BioC-compliant architecture well positioned for use by Track 3.

### *Training Phase*

In order for participants to gain an understanding of CTD curation and associated NER requirements, participants were provided with a comprehensive set of training materials in May 2013. A detailed document entitled *Summary Of Curation Details For The Comparative Toxicogenomics Database*, was distributed to participants ([https://gillnet.mdibl.org/~twiegers/bciv/CTD\\_curation\\_summary.docx](https://gillnet.mdibl.org/~twiegers/bciv/CTD_curation_summary.docx)). In addition, a training dataset was made available that consisted of 1,112 articles previously manually curated by CTD biocurators. The training dataset was provided in a single BioC XML-based file (<https://gillnet.mdibl.org/~twiegers/bciv/bcIVLearningCorpus.xml>), and contained important details associated with the articles in the dataset, including the PubMed ID, title, abstract, gene, chemical, disease, and action term annotations, and a list of associated curated interactions. References to general BioC information, BioC DTDs, and a key file that described the BioC XML format in the context of the learning corpus, were also provided to participants, as were sample Web service requests and responses for each NER category. Finally, the complete CTD controlled vocabularies, in multiple formats and including both terms and synonyms, were provided for each of the NER categories.

During July 2013, the BioCreative IV Track 3 NER Testing Facility Web site was released (Figure 1). This testing facility provided a front-end to a CTD Web service that upon execution called the participant's Web service, enabling participants to test their Web services against the training dataset. The participants simply entered a PubMed ID (from the training dataset), the URL of their Web service, an NER type (*i.e.*, gene, chemical, disease, or action term), and report format-related information,. This would cause CTD's Web service to call the participant's Web service using BioC XML for inter-process communications; CTD's Web service would in turn receive text-mined annotations from the participant's Web service using BioC XML. CTD's Web service would then process the annotations and compute the results, providing the user with recall, precision, response time, and a detailed list of curated terms, text-mined terms, and text-mined term hits. The participants were also given the opportunity to bypass the Web-based front-end and call the CTD Web service directly via application-to-application HTTP POST calls; this feature enabled users to run batch processes against the entire training dataset. The testing facility was heavily used, receiving over 260,000 hits from its inception through completion of the training phase.



**Figure 1.** Participants were provided with the *BioCreative IV Track 3 NER Testing Facility* developed by CTD to enable participants to test their NER Web services.

### ***Testing Phase and Methodology***

On August 19, 2013, participants were asked to submit to Track 3 organizers Web service URLs for testing, as well as brief system descriptions. CTD staff then tested the Web services against a test dataset of 510 articles manually curated by CTD staff using a client process specifically developed for testing. The process tested one abstract at a time, and participants were unaware of the articles to be tested prior to testing. The test dataset included 1,122 distinct curated genes, 1,192 chemicals, 943 diseases, and 966 chemical/gene-specific action terms, all within the context of 3,953 manually curated interactions.

Recall, precision, balanced F-score (sometimes referred to as F1 score or F-measure), and response times were captured for each Web service call. Recall scores were calculated by dividing the number of distinct curated actors identified by the text-mining tools—either by a synonym to the term or by the term itself—by the total number of distinct curated actors. Precision scores were calculated by dividing the number of distinct curated actors identified by the text-mining tools by the number of distinct text-mined terms. Balanced F-scores were calculated as follows:

$$\text{Balanced F-score} = 2 * ((\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision}))$$

Response times were calculated by measuring the duration between the call from CTD to the respective Web service, and receipt of communications back to CTD from the Web service. All testing was performed by CTD's software developer. Micro-averaging was used for aggregate recall, precision, F-score, and response time.

It is important to note that the standard text-mining metric calculations of precision and recall may be imperfect within the context of CTD curation. The gold standard data were comprised of curated—rather than cited—gene/protein, disease and chemical/drug actors within each abstract. There are likely to be instances where valid, cited actors are not actually involved in the types of interactions captured by CTD curators; furthermore, there are instances where curated actors are identified by curators only in the full text of the article. Consequently, the complete universe of valid and cited actors specifically resident within each abstract is not recorded by CTD curators and is therefore unknown.

### **Results and Discussion**

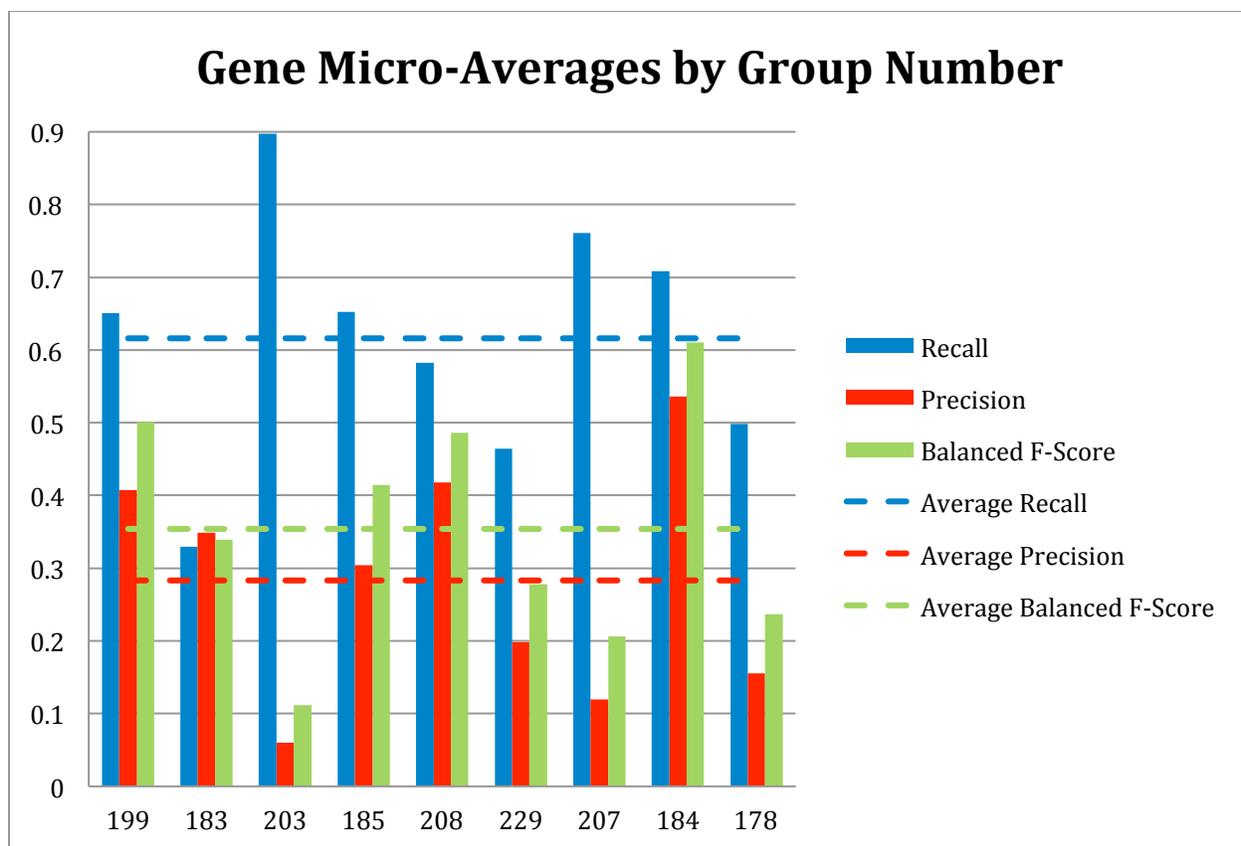
A total of 12 groups participated:

<b>Institution</b>	<b>Department/Division</b>	<b>Location</b>
National ICT Australia	Victoria Research Laboratory	Melbourne, Australia
Wuhan University		Wuhan, Hubei, China
University of Applied Sciences University and University Hospitals of Geneva	BiTeM Group, Information Science Department Division of Medical Information Sciences	Geneva, Switzerland Geneva, Switzerland
SIB Swiss Institute of Bioinformatics	SIBtex	Geneva, Switzerland
University of Zurich	Institute of Computational Linguistics	Zurich, Switzerland
Academia Sinica	Institute of Information Science	Taipei, Taiwan
Yuan Ze University	Department of Computer Science & Engineering	Taoyuan, Taiwan
Taipei Medical University	Graduate Institute of BioMedical Informatics	Taipei, Taiwan
National Tsing-Hua University	Department of Computer Science	HsinChu, Taiwan
National Central University	Department of Computer Science and Information Engineering	Zhongli City, Taiwan
National Cheng Kung University (2)	Department of Computer Science and Information Engineering	Tainan, Taiwan
Mayo Clinic	Department of Health Sciences Research	Rochester, MN, USA
OntoChem GmbH		Halle/Saale, Germany
University of Manchester	National Centre for Text Mining	Manchester, United Kingdom
RelAgent Pvt Ltd		Adyar, Chennai, India
Anna University	AU-KBC Research Centre	Chrompet, Chennai, India

These groups submitted a combined total of 44 Web services for testing. Of the 44 Web services submitted, 39 were successfully tested against the complete test dataset and included nine gene-, ten chemical, ten disease-, and ten action term-based NER Web services. The remaining five Web services were fully operational, but were unable to process the complete test dataset for varying reasons. In three of the five cases, it appeared as though indexing of the complete PubMed corpus was necessary prior to processing, and some of the PubMed abstracts in the test dataset had not yet been indexed. The reasons for failure of the remaining Web services were unclear.

### ***Gene/Protein NER Results***

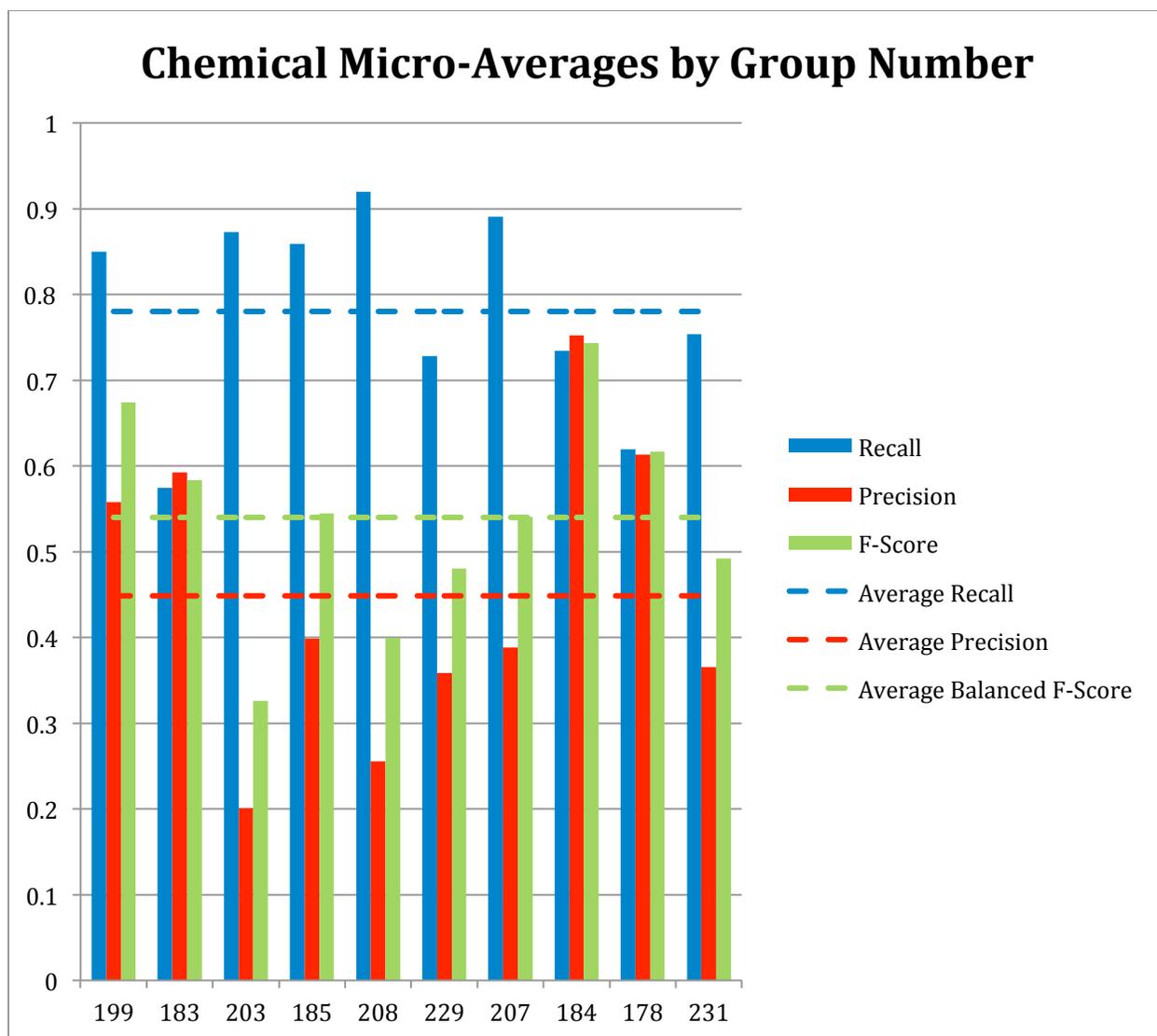
Among the 12 submissions for gene/protein NER, nine were successfully tested. As shown in Figure 2, average recall was 62% and ranged from 32% to 89%. Average precision was 28% and ranged from 6% to 54%. Average balanced F-scores were 36% and ranged from 11% to 61%. Interestingly, the two groups with the highest recall scores also had the lowest precision and F-scores, suggesting a much stronger emphasis on recall at the expense of precision. The average response time was 9.3 seconds and ranged from 0.14 to 61 seconds, with a large standard deviation of 19.6 (Figure 7).



**Figure 2.** Gene recall (blue), precision (red), and balanced F-score (green) results are shown for each participating group (anonymously identified by group number). Average scores for each metric (dotted lines) are also provided.

### ***Chemical/Drug NER Results***

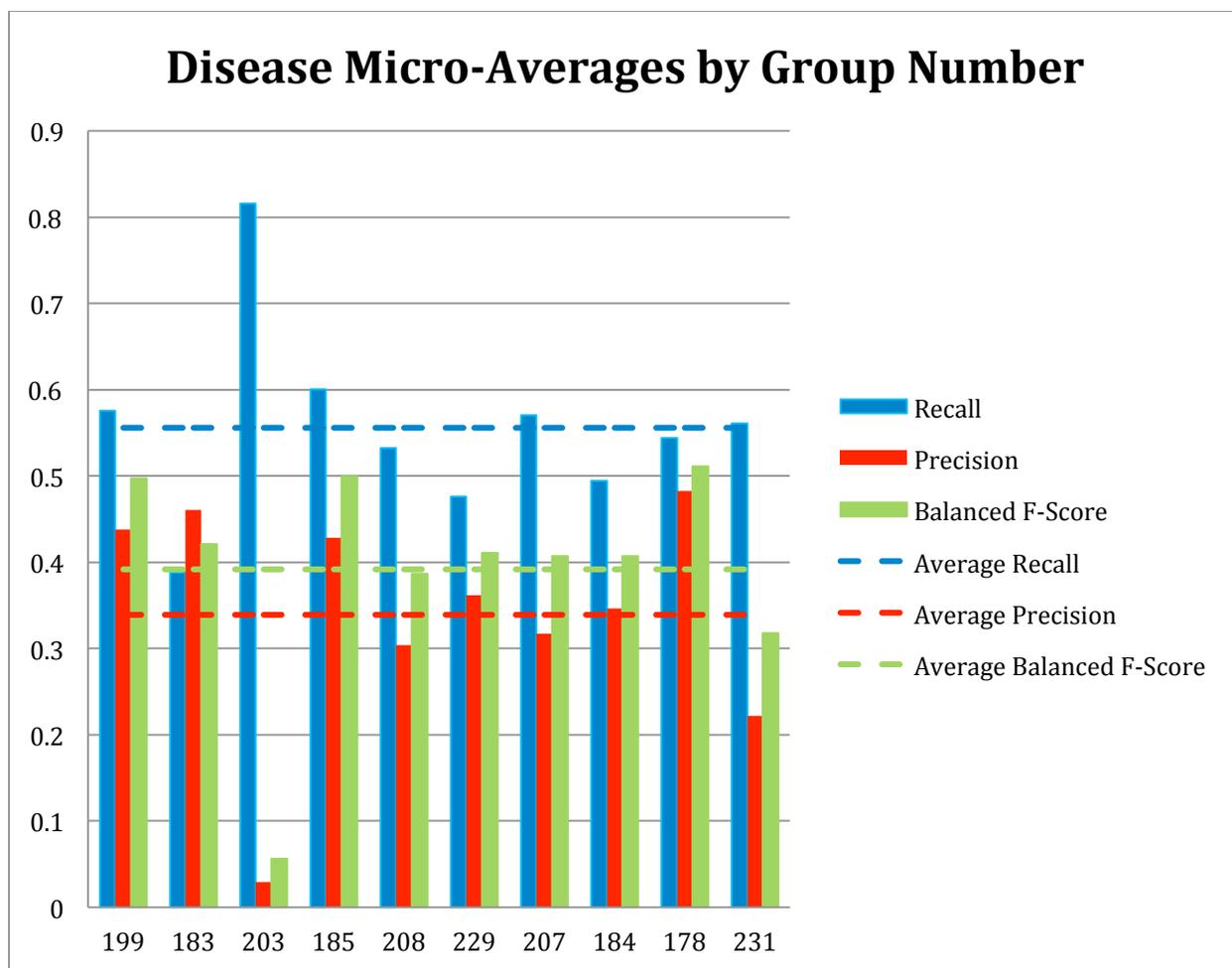
Among the 11 submissions for chemical/drug NER, 10 were successfully tested (Figure 3). Average recall was 78% and ranged from 57% to 92%. Average precision was 45% and ranged from 20% to 75%. Average balanced F-scores were 54% and ranged from 33% to 74%. The average response time was 4.7 seconds and ranged from 0.14 to 27 seconds, with a standard deviation of 8.8 (Figure 7).



**Figure 3.** Chemical recall (blue), precision (red), and balanced F-score (green) results are shown for each participating group (anonymously identified by group number). Average scores for each metric (dotted lines) are also provided.

### ***Disease NER Results***

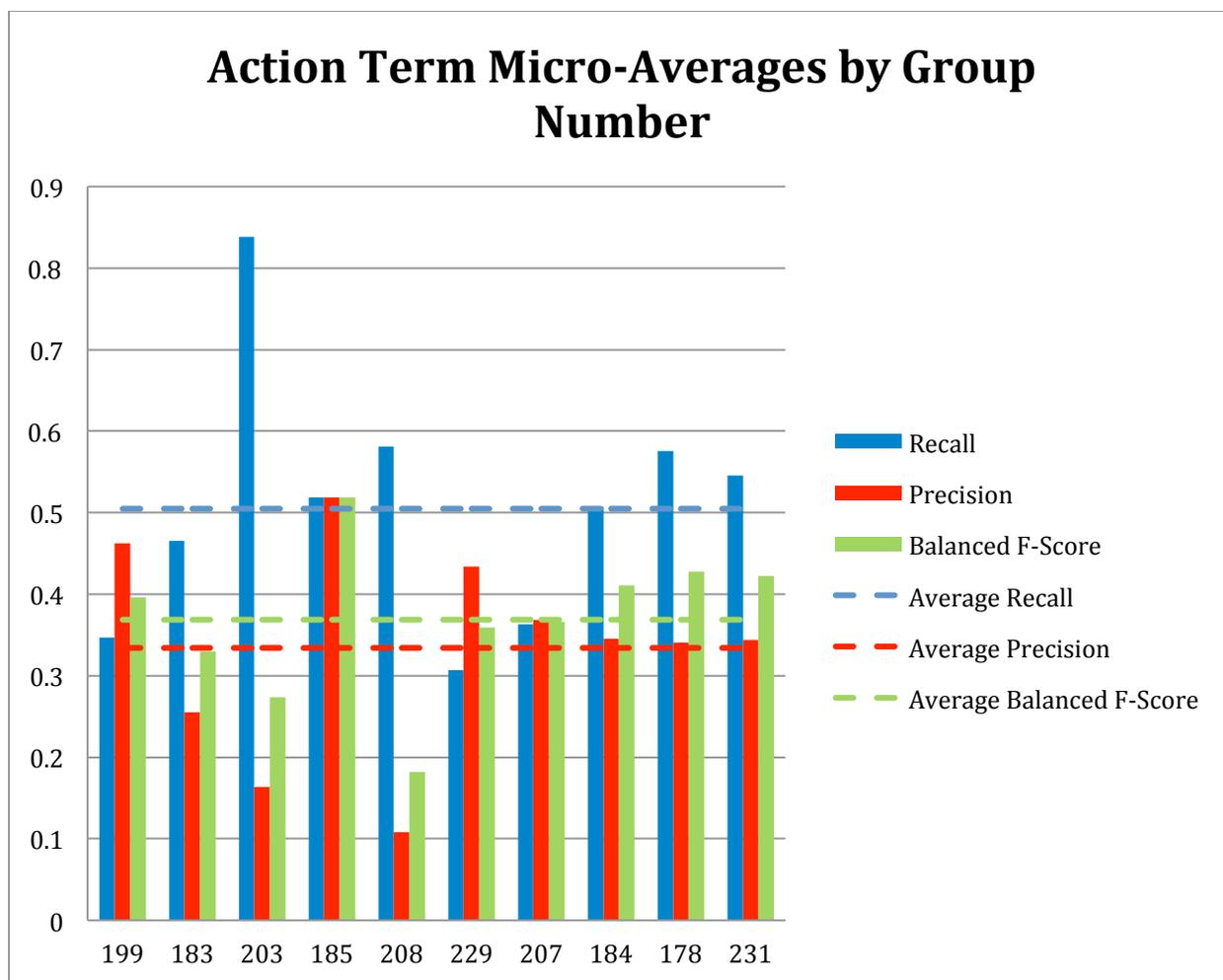
Ten of the 11 submissions for disease NER were successfully tested (Figure 4). Average recall was 56% and ranged from 39% to 82%. Average precision was 34% and ranged from 3% to 48%. Average balanced F-scores were 39% and ranged from 6% to 51%. The average response time was 2.9 seconds and ranged from 0.13 to 22 seconds, with a standard deviation of 6.6 (Figure 7).



**Figure 4.** Disease recall (blue), precision (red), and balanced F-score (green) results are shown for each participating group (anonymously identified by group number). Average scores for each metric (dotted lines) are also provided.

#### ***Chemical/Gene Action Term NER Results***

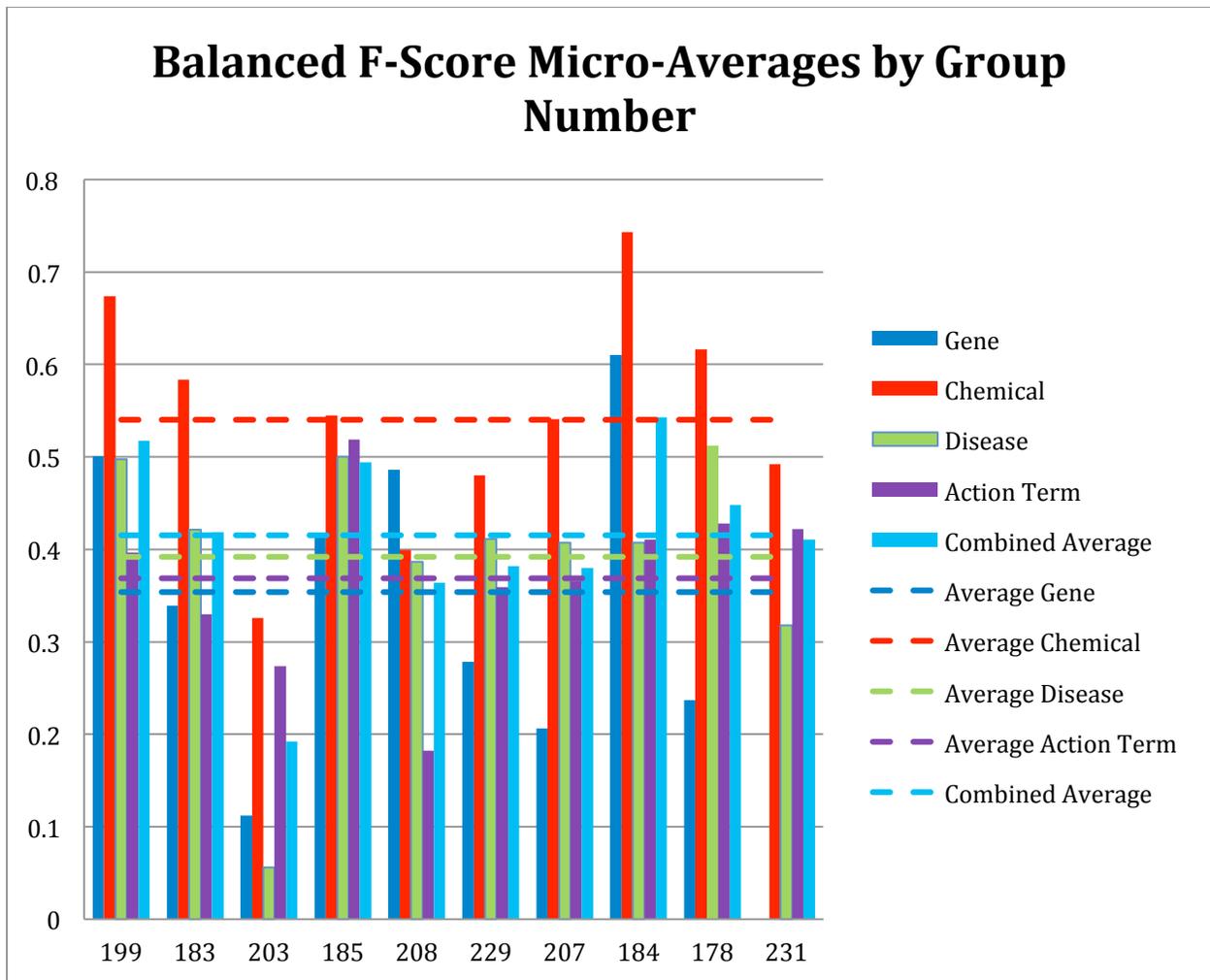
All of the 10 submissions for chemical/gene action NER were successfully tested (Figure 5). Average recall was 50% and ranged from 31% to 84%. Average precision was 33% and ranged from 11% to 52%. Average balanced F-scores were 37% and ranged from 18% to 52%. The average response time was 5.1 seconds and ranged from 0.13 to 24 seconds, with a standard deviation of 9.4 (Figure 7).



**Figure 5.** Chemical/gene action term recall (blue), precision (red), and balanced F-score (green) results are shown for each participating group (anonymously identified by group number). Average scores for each metric (dotted lines) are also provided.

#### *Aggregate F-Score Results*

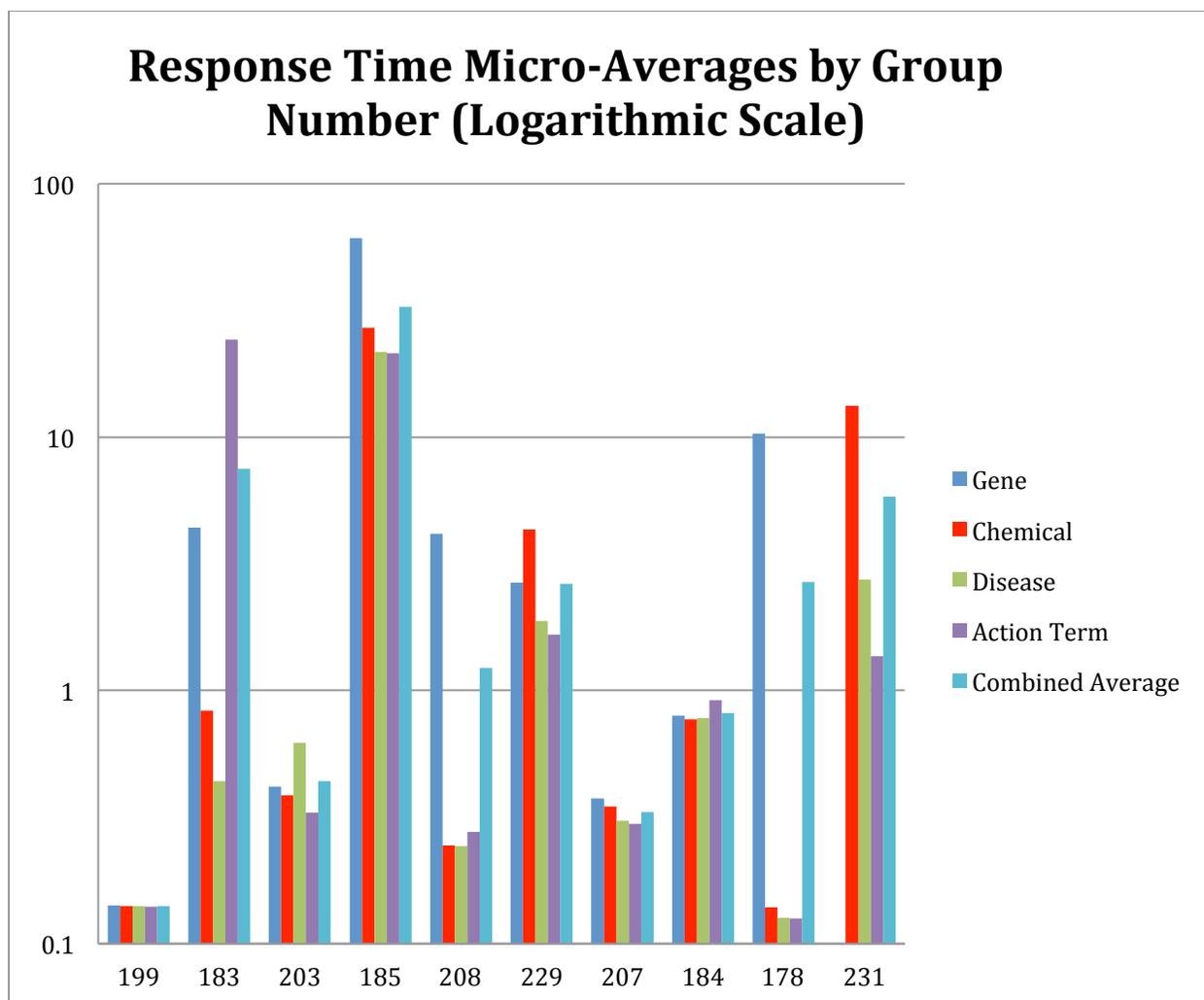
A total of ten groups submitted 39 Web services that were successfully tested. Average combined balanced F-score was 41% and ranged from 19% to 54% (Figure 6).



**Figure 6.** Balanced F-score results for each NER category, as well as a combined average, are provided for each participating group (anonymously identified by group number). Average scores for each metric (dotted lines) are also provided.

### ***Response Time Results***

The combined average response times across NER categories was 5.4 seconds, with a standard deviation of 9.9 (Figure 7). The fastest combined average response time was 0.14 seconds, and the slowest was 32.8 seconds. The variability appeared to be a function of the NER process itself and/or the hardware on which the process was run; there was no apparent correlation either to geographic area or NER recall or precision performance.



**Figure 7.** Response time results for each NER category, as well as a combined average, are provided for each participating group (which are anonymously identified by group number). Note: a logarithmic scale is used.

**Discussion**

The results of Track 3 testing clearly validate the conceptual feasibility of integrating Web service-based NER functionality into asynchronous, batch-oriented text-mining pipelines. The participant NER performance and Web service response times were more than adequate for such use in most cases. In making these assertions it must be emphasized that equal weight was placed on response times as was placed on NER performance in evaluating Track 3. Although response times might seem secondary to NER performance, a limiting factor to remote processing of the type proposed here is clearly response time: if the best NER tools have inadequate response times, they are of little use to resources like CTD. At times the CTD text-mining pipeline processes tens of thousands of abstracts; poor response times can equate to days to process large datasets even in cases where multi-threading is introduced. For example, the

worst gene NER response time was one minute, which for a 6,000 article dataset equates to 100 hours of sequential processing, whereas a response time of five seconds equates to a much more manageable 8.3 hours of sequential processing.

Among the most positive findings of Track 3 is that neither geographic location nor NER performance appeared to play a role in response times; some of the best NER-performing groups had the fastest response times. Group 184, for example, delivered strong performance in every NER category with fraction-of-a-second response times. Group 199 delivered similar results with even faster response times. And some of the more remote groups provided among the fastest response times (specifics are not provided to maintain group anonymity). One of the group's response time was largely dependent upon whether the article in question had been indexed; once the article had been indexed, the response time moved from in excess of 10 seconds down to fractions of a second for the remaining major NER types. Even the group with the slowest response times, 185, delivered above average NER recall and precision performance and attributed their slow response times to inferior hardware rather than anything inherent to the NER process itself. From a conceptual perspective, the results were very encouraging, providing a proof-of-concept that Web services-based NER is feasible for asynchronous processing. BioC proved to be an extremely robust, effective tool in standardizing high level inter-process communications. The framework provided all the functionality required for Track 3, and did so in a very unobtrusive fashion: the vast majority of the participants required little if any help from the organizers with respect to BioC, there were few syntax errors in the BioC XML returned from the Web services, and those syntax errors that did exist for the most part were quickly corrected. The fact that CTD did not have to create an application-specific inter-process communications framework was very beneficial to the track, and in the end the tools developed for the track provided a level of interoperability that would not have otherwise existed in the absence of BioC.

The REST-compliant architecture also proved to be an excellent design choice because its design goal to abstract the architectural elements of distributed systems [16] complemented the Track 3 goal. In addition, the fact that 39 fully functional NER Web services were set up within a few months, communicating via a common framework, speaks to the usability of the REST architecture style.

The NER Testing Facility Web site was heavily used and very successful in ensuring that the groups were producing Web services that were in compliance with the requirements established for Track 3. The fact that all 12 groups provided functional Web services (if not fully functional against the entire test dataset for two of the sites) lends support to the effectiveness of the testing facility. The site provided a simple way to test applications during the training phase of the project to ensure the correct syntax. The feature that enabled users to bypass the Web-based front-end and call the CTD Web service directly via application-to-application HTTP POST calls

provided participants with the ability to refine their NER performance against the entire training dataset.

Chemicals/drugs continue to be the strongest interaction actor NER recall category on average (78%), mirroring past CTD-related studies [7, 8, 13], followed by genes (62%) and diseases (56%). The performance of action terms NER was far superior to the BioCreative 2012 results, but continued to be somewhat disappointing; although the top recall score was 84%, the precision for that score was 11%, the lowest in the group. Since the population of potential values is very small for action terms in relation to the other NER categories, much greater emphasis is placed on precision scores, and the 33% average precision score was disappointing. One exception was Group 185, which scored over 50% in both recall and precision. Action term recognition is an extremely important area for CTD in that the ability to accurately recognize these terms would significantly improve CTD's ability to identify articles that contain curatable information. More modeling, analysis, and design work needs to be done by CTD, either internally or in conjunction with other collaborators, in order to develop NER algorithms and tools to better recognize actions terms when they appear in text.

Although CTD has not tested its own tools against the test dataset, plans are underway to do so. Preliminary testing using a pared-down version of gene NER placed CTD's tools in the middle of the gene group in recall, precision, and F-score. Where possible, CTD aims to collaborate with participants who developed better performing tools to further extend the impact of this project. In conclusion, the results of Track 3 underscores the extraordinary ability of Web services to abstract from users the complexity of underlying computational systems, freeing them to focus on performance. The fact that Track 3 has proven to be so successful brings with it the possibility that text-mining groups like CTD could mix-and-match NER functionality based solely on the expertise and performance of the NER provider, rather than on the characteristics of the respective tool's underlying technical architecture or geographic locale.

## **Funding**

This program is supported by funds from the National Institute of Environmental Health Sciences (ES014065). The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

*Conflict of interest.* None declared.

## References

1. Davis, A.P., Murphy, C.G., Johnson, R., *et al.* (2013) The Comparative Toxicogenomics Database: update 2013. *Nucleic Acids Res.*, 41, D1104-D1114.
2. Davis, A.P., Wieggers, T.C., Murphy, C.G., *et al.* (2011) The curation paradigm and application tool used for manual curation of the scientific literature at the Comparative Toxicogenomics Database. *Database*, Sep 20;2011:bar034.
3. Davis, A.P., Wieggers, T.C., Rosenstein, M.C., *et al.* (2012) MEDIC: a practical disease vocabulary used at the Comparative Toxicogenomics Database. *Database*, Mar 20;2012:bar065.
4. Amberger, J., Bocchini, C., and Hamosh, A. (2011) A new face and new challenges for online mendelian inheritance in man (omim(r)). *Hum Mutat.* 32, 564-567.
5. Coletti, M.H. and Bleich, H.L. (2001) Medical Subject Headings used to search the biomedical literature. *J Am Med Inform Assoc.*, 8, 317-323.
6. Maglott, D., Ostell, J., Pruitt, K.D., and Tatusova, T. (2011) Entrez gene: Gene-centered information at ncbi. *Nucleic Acids Res.* 39, D52-57.
7. Wieggers, T.C., Davis, A.P., Cohen, K.B., *et al.* (2009) Text mining and manual curation of chemical-gene-disease networks for the Comparative Toxicogenomics Database (CTD). *BMC Bioinformatics*, 10, 326.
8. Davis, A.P., Wieggers, T.C., Johnson, R.J., *et al.* (2013) Text mining effectively scores and ranks the literature for improving chemical-gene-disease curation at the Comparative Toxicogenomics Database. *PLoS ONE*, 8, e58201.
9. Settles B. (2005) ABNER: an open source tool for automatically tagging genes, proteins and other entity names in text. *Bioinformatics*, 21(14):3191-3192.
10. Corbett P, Copestake A: Cascaded Classifiers for Confidence-Based Chemical Named Entity Recognition. *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing* 2008.
11. Corbett P, Murray-Rust P: High-throughput identification of chemistry in life science texts. In *Computational Life Sciences II. Volume 4216*. Heidelberg: Springer Berlin; 2006.
12. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. *Proc AMIA Symp* 2001, :17-21.
13. Wieggers, T.C., Davis A.P., Mattingly, C.J. (2012) Collaborative biocuration-text-mining development task for document prioritization for curation. *Database*, Nov 22;2012:bas037.
14. W3C Working Group, *Web Services Glossary*. W3C, Feb 11; 2006.
15. BioC: A Minimalist Approach to Interoperability for Biomedical Text Processing, *Database*, Sep 19;2013:bar064.
16. Fielding, Roy Thomas, (2000) Architectural Styles and the Design of Network-based Software Architectures, Doctoral dissertation, University of California, Irvine
17. W3C Working Group, *Web Services Architecture*. W3C, Feb 11; 2004.

# NaCTeM CTD Web Services

Riza Batista-Navarro\*, Rafal Rak and Sophia Ananiadou

National Centre for Text Mining, University of Manchester, UK

\*Corresponding author: Tel: +441613063091, E-mail: riza.batista-navarro@cs.man.ac.uk

## Abstract

Participating in the BioCreative IV CTD Curation shared task, we developed RESTful, BioC-compliant web services which recognise CTD chemicals, genes, diseases and actions terms in PubMed abstracts. The tools are based on machine learning approaches, specifically, conditional random fields (CRFs) for recognising names of chemicals, genes and diseases, and support vector machines (SVMs) for recognising action terms.

**Keywords:** Named entity recognition, Multiclass multilabel classification, Interoperability, Comparative Toxicogenomics Database, Curation

## Introduction

The Comparative Toxicogenomics Database (1) (CTD) is a publicly available resource that integrates information on chemicals, genes and diseases curated from scientific literature, aiming to foster understanding of the means by which drugs and chemicals affect human health. Relationships between entities (e.g., chemical-gene, chemical-disease and gene-disease) are stored in the database by means of manual curation. The BioCreative IV Track 3 was defined to encourage the text mining community to develop interoperable automatic tools to assist in CTD curation.

The task asks for preparing RESTful web services capable of accepting input in BioC format and responding with an enriched version that includes one of four concept types, namely, chemicals, genes, diseases and action terms.

We address this challenge by using machine learning-based approaches. Specifically, we used algorithms for sequence labelling (for identifying chemicals, genes and diseases) and multiclass, multilabel classification (for identifying action terms), whilst leveraging relevant resources such as the CTD vocabularies and other ontologies/dictionaries.

## Methods

In this section, we provide an overview of the various methods, as well as resources, which were used in the development of the CTD automatic curation tools.

## Resources

The organisers provided a training corpus of 1,112 PubMed abstracts encoded in the BioC XML format. Each abstract consists of a list of manually curated chemicals, genes, diseases and action terms. The annotations do not include specific textual locations of the concepts, and they correspond to the preferred names of the concepts in the CTD vocabularies, rather than the surface forms appearing in actual text.

The automatic annotation methods described in the following sections heavily rely on several external dictionaries. Apart from the chemical, gene and disease vocabularies available in CTD, we also used 1) Chemical Entities of Biological Interest (2), DrugBank (3), Joint Chemical Dictionary (4), and PubChem Compound (5) for chemicals, 2) UniProt (6), NCBI EntrezGene (7), GeneLexicon (8), and Human Genome Organisation Ontology (9) for genes, and 3) Medical Subject Headings (10), Unified Medical Language System (11), Disease Ontology (12), and Online Mendelian Inheritance in Man Ontology (13) for diseases.

## Chemicals, Genes and Diseases

We cast the problem of recognising chemicals, genes and diseases as a named entity recognition (NER) task; specifically, we modelled the data using conditional random fields (14) (CRFs).

*Development phase.* Since the training corpus does not contain the locations of annotations, the first challenge we addressed was the generation of a silver-annotated corpus for each entity type. Using the dictionaries listed in the previous section, we extracted the locations of chemicals, genes and diseases in the abstracts in the provided training corpus. This, however, introduced a considerable amount of noise due to the ambiguity of certain names (e.g., the chemical *lead* matches verbs of the same form). To mitigate this problem, we exploited the testing facility provided by the Track 3 organisers to identify false positives and filter them out. The true positives were then used in silver-annotating the corpora with the specific locations of entities in text.

We observed, however, that in silver-annotating the corpus for diseases, many of the names in the gold standard annotations were missed due to the various ways in which they are expressed in text. For example, the name *leukopenia* appears as a curated disease for an abstract. Whilst the adjective *leukopenic* appears in text, the name itself (nor any of its synonyms) does not. To capture such cases, we developed a heuristic method based on overlapping stemmed tokens. This method is based on the following preliminary steps, performed on both the dictionary entries in the CTD disease vocabulary as well as the noun phrases in text: 1) removal of stop words, 2) stemming of each remaining token and 3) alphabetical reordering of tokens. For each noun phrase-dictionary entry pair, a score is computed based on the number of common tokens. If the score is greater than a threshold, the matching tokens are silver-annotated in text.

Utilising the NERSuite package (15), we trained a CRF model for each of the categories using the silver-annotated corpora. We exploited lexical, orthographic, syntactic, and dictionary features.

*Extraction phase.* Each input abstract is pre-processed by splitting sentences (using the MEDLINE sentence model in LingPipe (16)), tokenisation (using OSCAR4 (17)), and part-of-speech and chunk tagging (using GENIA Tagger (18)). NERSuite then generates the features and assigns labels to the token sequences using the trained models. If any of the named entities tagged by the model cannot be mapped to a CTD preferred name, we apply the previously mentioned heuristic method based on overlapping stemmed tokens to retrieve and return the CTD entry with the highest score.

### **Action Terms**

Initially, we treated the recognition of action words the same way as the other categories, i.e., as sequence labelling problem. However, unlike chemicals, genes and diseases, CTD action terms are expressed in text much less explicitly. Action terms such as *response to substance* would very rarely appear verbatim in actual text, with authors expressing the same idea by instead saying that *A affects B in some manner C*. For this reason, and considering that there are only 53 possible CTD action terms, we decided to cast the problem as a multiclass, multilabel classification task, i.e., each abstract may be classified with any number of action terms depending on the types of chemical-gene interactions that particular abstract pertains to.

*Development Phase.* Employing a one-versus-all approach, we used support vector machines (SVMs) (19) to train a total of 53 different models (i.e., one for each of the 53 CTD action terms). The feature set included 1) verb variant matching based on BioLexicon (20), and 2) the co-occurrence (and proximity) of chemical and gene names with a biomedical verb variant. Features of the first type are represented as booleans, while those of the second type are normalised weights accumulated based on the number of co-occurrences.

*Extraction Phase.* Each input abstract undergoes the same pre-processing pipeline as used for the other categories described in the previous section. Chemical and gene names (needed as features for the classification) are extracted automatically using the models described in the previous section. If the prediction returned by any of the 53 models is greater than a threshold, the document is labelled with the CTD action term corresponding to that model.

### **Availability**

The web services follow the specification set by the task organisers and are accessible at the following locations:

Chemicals: <http://nactem.ac.uk/CTDWebService/ctd/chem>

Genes: <http://nactem.ac.uk/CTDWebService/ctd/gene>

Diseases: <http://nactem.ac.uk/CTDWebService/ctd/disease>

Action terms: [http://nactem.ac.uk/CTDWebService/ctd/action\\_term](http://nactem.ac.uk/CTDWebService/ctd/action_term)

## Funding

This work was partially supported by Europe PubMed Central funders (led by Wellcome Trust).

## References

1. Davis, A.P., et al., *The Comparative Toxicogenomics Database: update 2013*. Nucleic Acids Res., 2013. **41**(D1): p. D1104-D1114.
2. Hastings, J., et al., *The ChEBI reference database and ontology for biologically relevant chemistry: enhancements for 2013*. Nucleic Acids Res., 2013. **41**(D1): p. D456-D463.
3. Knox, C., et al., *DrugBank 3.0: a comprehensive resource for 'Omics' research on drugs*. Nucleic Acids Res., 2011. **39**(suppl 1): p. D1035-D1041.
4. Hettne, K.M., et al., *A dictionary to identify small molecules and drugs in free text*. Bioinformatics, 2009. **25**(22): p. 2983-2991.
5. Bolton, E.E., et al., *PubChem: Integrated Platform of Small Molecules and Biological Activities*. Annu. Rep. Comput. Chem., 2008. **4**.
6. Consortium, T.U., *Update on activities at the Universal Protein Resource (UniProt) in 2013*. Nucleic Acids Res., 2013. **41**(D1): p. D43-D47.
7. Maglott, D., et al., *Entrez Gene: gene-centered information at NCBI*. Nucleic Acids Res., 2007. **35**(suppl 1): p. D26-D31.
8. Wilbur, W.J. and L. Tanabe, *Generation of a large gene/protein lexicon by morphological pattern analysis*. J. Bioinf. Comput. Biol., 2004. **01**(04): p. 611-626.
9. Gray, K.A., et al., *Genenames.org: the HGNC resources in 2013*. Nucleic Acids Res., 2013. **41**(D1): p. D545-D552.
10. Nelson, S.J. *Medical Terminologies That Work: The Example of MeSH*. In Proceedings: 10th International Symposium on Pervasive Systems, Algorithms, and Networks (ISPAN) 2009 2009.
11. Bodenreider, O., *The Unified Medical Language System (UMLS): integrating biomedical terminology*. Nucleic Acids Res., 2004. **32**(suppl 1): p. D267-D270.
12. Schriml, L.M., et al., *Disease Ontology: a backbone for disease semantic integration*. Nucleic Acids Res., 2012. **40**(D1): p. D940-D946.
13. Hamosh, A., et al., *Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders*. Nucleic Acids Res., 2005. **33**(suppl 1): p. D514-D517.
14. Lafferty, J.D., A. McCallum, and F.C.N. Pereira. *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data*. In Proceedings: Eighteenth International Conference on Machine Learning. 2001. Morgan Kaufmann Publishers Inc.
15. Cho, H.-C. *NERsuite: A Named Entity Recognition toolkit*. 2012; Available from: <http://nersuite.nlplab.org/>.
16. Alias-i. *LingPipe 4.1.0*. 2008; Available from: <http://alias-i.com/lingpipe>.

17. Jessop, D., et al., *OSCAR4: a flexible architecture for chemical text-mining*. J. Cheminf., 2011. **3**(1): p. 41.
18. Tsuruoka, Y., et al. *Developing a Robust Part-of-Speech Tagger for Biomedical Text*. In Proceedings: *Advances in Informatics - 10th Panhellenic Conference on Informatics*. 2005. Springer-Verlag.
19. Joachims, T., *Learning to Classify Text Using Support Vector Machines. Methods, Theory and Algorithms*. Vol. 668. 2002: Springer.
20. Thompson, P., et al., *The BioLexicon: a large-scale terminological resource for biomedical text mining*. BMC Bioinf., 2011. **12**: p. 397.

# OntoGene: CTD entity and action term recognition

Fabio Rinaldi, Simon Clematide, Tilia Renate Ellendorff, Hernani Marques

## Motivation and Objectives

Text mining technologies are increasingly providing an effective response to the growing demand for faster access to the vast amounts of information hidden in the literature. Several tools are becoming available which offer the capability to mine the literature for specific information, such as for example protein-protein interactions or drug-disease relationships. The biomedical text mining community regularly verifies the progress of such systems through competitive evaluations, such as BioCreative, BioNLP, i2b2, CALBC, CLEF-ER, BioASQ, etc.

The OntoGene system is a text mining system which specializes in the detection of entities and relationships from selected categories, such as proteins, genes, drugs, diseases, chemicals. The quality of the system has been verified several times through participation in some of the community-organized evaluation campaigns, where OntoGene has consistently achieved top-ranked results. Some highlights include best results in the detection of experimental methods (BioCreative 2006), best results in the detection of protein-protein interactions (BioCreative 2009), best results in large-scale detection of some categories of biomedical entities (CALBC 2010), best overall results in the CTD triage task (BioCreative 2012).

However, the OntoGene system is based on a relatively heterogeneous pipeline, which would not be easily portable to other sites. In order to make the advanced text mining capabilities of the OntoGene system more widely accessible to the biomedical community without the burden of installation of complex software, we long planned to provide access through web services.

The task 3 of BioCreative 2013 provided the ideal setting to implement an initial version of such web service interface. The goal of task 3 was to deliver entity and action term annotation for the Comparative Toxicogenomics Database (3).

## Methods

The text mining pipeline which constitutes the core of the OntoGene system has been described previously in a number of publications (5,6). We will only briefly describe the core text mining technologies, and instead focus mainly on the novel web service which allows remote access to the OntoGene text mining capabilities.

The first step in order to process a collection of biomedical literature consists in the annotation of names of relevant domain entities in biomedical literature (currently the systems considers

proteins, genes, species, experimental methods, cell lines, chemicals, drugs and diseases). These names are sourced from reference databases and are associated with their unique identifiers in those databases, thus allowing resolution of synonyms and cross-linking among different resources. A term normalization step is used to match the terms with their actual representation in the text, taking into account a number of possible surface variations. Finally, a disambiguation step resolves the ambiguity of the matched terms.

A supervised machine learning method is used to generate a score for entity annotations. Since the term recognizer aims at high recall, it introduces several noisy concepts, which we want to automatically identify in order to penalize them. Additionally, we need to adapt to highly-ranked false positive relations which are generated by our frequency based approach. The goal is to identify some global preference or biases which can be found in the reference database. Our technique is to weight individual concepts according to their likeliness to appear as an entity in a correct relation, as seen in the target database. The same approach was previously used for our participation in BioCreative 2012 (8). The only adaptation was to use the most recent version of the CTD datasets for training (about 97'000 pubmed articles), filtered by the criterion that there were not more than 12 relations curated in an article. This led to a number of 328230 curated relations from these articles where we applied the supervised distant learning approach for scoring the concept relevance. The term database for genes, chemicals and diseases has 454,429 concepts and 1,282,582 terms.

The OntoGene web service has been implemented as a RESTful service (2). It accepts simple XML files as input, based on the BioC specification<sup>1</sup>. The output of the system is generated in the same format. Options can be used in the input query to select whether the result should contain in-line annotations (showing where exactly in the text the term was mentioned), or stand-off annotations. Currently the system uses pre-defined terminology, however we foresee in future the possibility to upload own terminologies, or select which subsets of the available terminology should be used.

### **Action Terms**

In order to be able to discover action terms in unseen abstract, we built several binary machine-learning classifiers, one for each action-term type. We did not use the ontogene pipeline for building the classifiers, but decided to base our system mainly on tools from the natural language processing toolkit NLTK (1). As training material, we made use of the official CTD data for gene-chemical interaction which can be downloaded from the website in xml-format as well as the referenced abstracts from pubmed. In addition to the abstracts, we used the MeSH descriptors and qualifiers as PubMed metadata. Any preprocessing of the abstracts was not done, apart from sentence splitting and tokenization.

---

<sup>1</sup> <http://www.ncbi.nlm.nih.gov/CBBresearch/Dogan/BioC/>

For building classifiers which were able to classify abstracts according to the action terms which they contain, we used the Naive Bayes classifier module, provided by NLTK. For each action term to be classified, one binary classifier was built. All features used are independent binary features, as required by the Naive Bayes theorem. We programmed the classifiers in a way that made it possible to use different combinations of feature-types. The simplest feature-type only uses the words of the abstract (bag-of-words). Other feature-types considered the stems of the words, word-bigrams, stem-bigrams, mesh-descriptors and mesh-qualifiers. Furthermore, the number of most frequent features used for a feature-type could be adjusted. Experiments showed that using the 5000 most frequent features for each feature type (e.g. the 5000 most frequent words are used as features) leads to the best results.

Another setting that we varied in order to find optimal performance is the number of action term types for which classifiers were included. Out of a total of 53 action terms, we made experiments with systems including classifiers for from 7 to 15 action terms. The best performance in terms of F-Score could be measured for the system which included 9 different classifiers.

The last variable of the classifying system was the size of its training set which consisted of abstracts randomly chosen from the total number of pubmed abstracts listed in CTD. Here it is important to take the efficiency of the system into account: the classifier tends to run very slow if too much data is provided, without big improvement once a certain amount of data is reached. We found that a training set of 2000 different abstracts shows a reasonably good performance together with an adequate speed rate.

With the help of experiments using different feature settings, we determined the best choice of features as bag-of-words, stem bigrams and mesh descriptors. In this context we found that mesh descriptors are the most useful features for determining action words followed by stem-bigrams. (Even though word-bigrams were found to be still a bit more useful than bag-of-words, using both at the same time seems to introduces redundant information and leads to a worse performance, the same seems to be the case with using stems together with stem-bigrams.) Using Mesh-qualifiers together with Mesh-descriptors as one feature proved to be too sparse to have any positive effect.

## **Results and Discussion**

Users can submit arbitrary documents to the OntoGene mining service by embedding the text to be mined within a simple XML wrapper. Both input and output of the system are defined according to the BioC standard (2). However typical usages will involve processing of PubMed abstracts or PubMed Central full papers. In this case the user can provide as input simply the pubmed identifier of the article. Optionally the users can specify which type of output they would like to obtain: if entities, which entity types, and if relationships, which combination of types.



## References

1. Bird, Steven, Edward Loper and Ewan Klein (2009). *Natural Language Processing with Python*. O'Reilly Media Inc.
2. Donald C. Comeau, Rezarta Islamaj Doğan, Paolo Ciccarese, Kevin Bretonnel Cohen, Martin Krallinger, Florian Leitner, Zhiyong Lu, Yifang Peng, Fabio Rinaldi, Manabu Torii, Alfonso Valencia, Karin Verspoor, Thomas C. Wieggers, Cathy H. Wu, W. John Wilbur (2013). BioC: a minimalist approach to interoperability for biomedical text processing, *The Journal of Biological Databases and Curation* (2013), *bat064*, doi:10.1093/database/bat064, published online.
3. Davis AP, Murphy CG, Johnson R, Lay JM, Lennon-Hopkins K, Saraceni-Richards C, Sciaky D, King BL, Rosenstein MC, Wieggers TC, Mattingly CJ. The Comparative Toxicogenomics Database: update 2013. *Nucleic Acids Res.* 2013 Jan 1;41(D1):D1104-14.
4. Richardson, Leonard; Ruby, Sam (2007), *RESTful Web Services*, O'Reilly, ISBN 978-0-596-52926-0
5. Fabio Rinaldi, Thomas Kappeler, Kaarel Kaljurand, Gerold Schneider, Manfred Klenner, Simon Clematide, Michael Hess, Jean-Marc von Allmen, Pierre Parisot, Martin Romacker, Therese Vachon (2008). OntoGene in BioCreative II. *Genome Biology*, 2008, 9:S13, PMC2559984
6. Fabio Rinaldi, Gerold Schneider, Kaarel Kaljurand, Simon Clematide, Thérèse Vachon, Martin Romacker, "OntoGene in BioCreative II.5," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 7(3), pp. 472-480, 2010. doi:10.1109/TCBB.2010.50
7. Fabio Rinaldi, Simon Clematide, Yael Garten, Michelle Whirl-Carrillo, Li Gong, Joan M. Hebert, Katrin Sangkuhl, Caroline F. Thorn, Teri E. Klein, Russ B. Altman. *Using ODIN for a PharmGKB revalidation experiment. The Journal of Biological Databases and Curation*, Oxford Journals, 2012, bas021; doi:10.1093/database/bas021
8. Fabio Rinaldi and Simon Clematide and Simon Hafner and Gerold Schneider and Gintare Grigonyte and Martin Romacker and Therese Vachon. Using the OntoGene pipeline for the triage task of BioCreative 2012. *The Journal of Biological Databases and Curation*, Oxford Journals, 2013. doi:10.1093/database/bas053

# Performance of a multi-class biomedical tagger on the BioCreative IV CTD task

S. V. Ramanan<sup>1,\*</sup> and P. Senthil Nathan<sup>1</sup>

<sup>1</sup>RelAgent Pvt Ltd, 56 Venkatratnam Nagar, Adyar, Chennai 600020, India

\*Corresponding author: Email: ramanan@npjoint.com

## Abstract

We have adapted Cocoa, an existing dictionary/rule based entity tagger that tags multiple semantic types in the biomedical domain to the BioCreative IV CTD task. We have added a normalization module against CTD dictionaries for the BioCreative CTD task. A preliminary evaluation of the system shows recall performance (82% for chemicals, 65% for diseases and 49% for genes) against the entire training corpus comparable to that achieved by other systems in the CTD task in BioCreative Workshop 2012 (1).

## Background

Cocoa (<http://npjoint.com/CocoaEval.html>) is an existing dictionary/rule based named entity tagger for the biomedical domain. The tagger simultaneously tags entities across a number of biomedical term classes, including chemicals, genes/proteins, biological/disease/generic processes and diseases. The tagger is already available through a web interface and also through RESTful APIs for a variety of formats, including the Genia A1 format. We have adapted this system to the BioCreative IV CTD task, primarily by adding an entity normalization module against CTD dictionaries. As required for the task, we have provided four endpoints for the various entity classes:

<http://npjoint.com/Cocoa/api/ctd/chem/>

<http://npjoint.com/Cocoa/api/ctd/gene/>

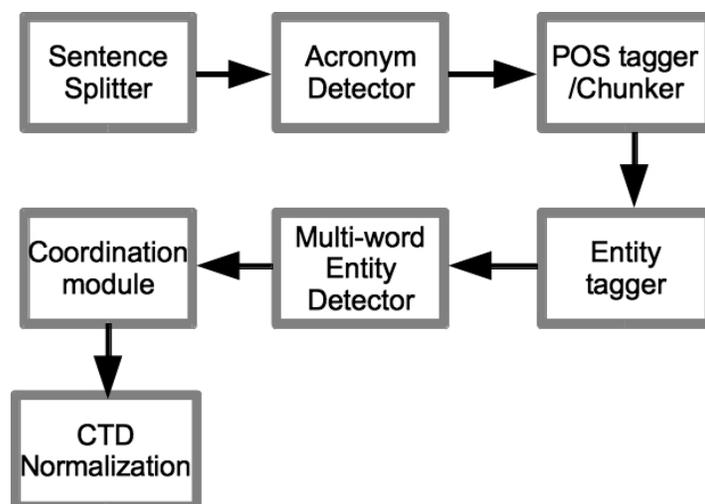
<http://npjoint.com/Cocoa/api/ctd/disease/>

[http://npjoint.com/Cocoa/api/ctd/action\\_term/](http://npjoint.com/Cocoa/api/ctd/action_term/)

## System description

The system consists primarily of six modules (see Fig 1):

(a) Sentence splitter. This uses various orthographical clues, in addition to periods, to detect the beginning of new sentences. Exceptions such as biological entities that begin in lower case letters (e.g., mRNA) are also accounted for.



**Figure 1.** Block Diagram of System

(b) Acronym detector. A dynamic programming algorithm is used to detect and tag acronyms.

(c) POS tagger/chunker. Both of these steps are handled by TBL-based algorithms in the public domain, namely the Brill tagger and fnTBL. The dictionaries and rules for these algorithms are heavily modified and tailored for the biomedical domain.

(d) Entity tagger. This module marks up single words in various entity classes. The approach primarily uses morphological clues for coverage ('-mab', '-ase', '-orrhea', '-tion'), mainly suffixes, while small dictionaries are used for terms not amenable to morphological analysis. Precision is increased by the use of false-positive dictionaries. The modules also marks up orthographically distinct entities (HYmA-1) as belonging to an 'unknown' entity class.

(e) Multi-word entity detector: Orthographically tagged entities in the 'unknown' class are merged with successor words known to belong to entity headword classes ('protein', 'kinase', 'symptom', 'disease'). A certain amount of word-sense disambiguation is also carried out in the module ('stones', 'solution', 'product', 'stem', 'relative', 'redness'). Common prefix terms for anatomical parts and diseases are merged here ('superior', 'severe', 'recurring').

(f) Coordination: Noun phrases in coordination are recognized in this module. Anatomical parts followed by disease headwords are merged ('liver cancer'); such mergers are also handled in coordination ('lung and liver cancer'). Terms in the 'unknown' class that can disambiguated through apposition are also marked up ('HDAC4, a member of the histone deacetylase family'). A formula detection submodule checks the remaining 'unknown' class terms for chemical formulae.

(g) Normalization: The modules described above are part of the existing web-accessible system, and were not modified for the CTD task, except for periodic upgrades as part of system maintenance and improvement. A new module for normalization against CTD dictionaries was however added specifically for the CTD task. For disease and chemicals, this was done primarily for exact matches against dictionary terms, after some small changes e.g. substitution of spaces with hyphens through regular expressions(' ' -> '[- ]'), matching against roughly synonymous headwords ['cancer' -> '(carcinoma|neoplasm|cancer)'] and irregular lemmatization ['renal' -> '(kidney|renal)']. For normalization of gene/protein terms, matches were re-attempted after stripping common headwords ('protein', 'gene', 'receptor') and by substitution of chemical premodifiers ('K' -> 'potassium' in 'K channel'). Action terms were handled, in addition to direct matches, by a certain amount of lemmatization as well as by synonyms, in addition to co-occurrences ('treatment' + 'response' -> 'response to substance').

### **Performance against training corpus**

As noted above, the entity-detection parts of the system were not specially modified for the CTD task. The system has performed reasonably against various corpora, and also in some shared tasks, such as the disease/sign /symptom recognition ShARe/CLEF eHealth task (F=0.78; test set); we also used the system-detected entities as an alternate to the gold-annotated entities for the BioNLP13 Cancer Genetics event extraction task without substantial changes in the performance(F=0.45 for event detection; test set).

Four modules to normalize entities against CTD dictionaries was the only addition for the CTD task. Match generalization against dictionaries through regular expressions was done by manually checking against about 50 abstracts in the training dataset. The modules were designed fairly conservatively to keep precision high even while boosting recall (which is the primary endpoint) for three entity categories (genes, chemicals and diseases). For the fourth category (action terms), the number of concepts is quite small (32), and we chose a strategy that increases recall only while maintaining or increasing the macro f-score.

We refined the normalization modules against a small portion (~5%) of the training corpus. The approximately performance against the entire training set is shown below in Table 1, and the official results (from the organizers) for the test set is also shown.

For chemicals and diseases, the results for disease and chemicals are similar for training and test sets. The recall against the test set for genes was considerably higher than for the training set, and this difference may be due the fact that we did not stringently check against synonyms (as is allowed by the CTD task) in our own evaluation method for the training set. The low numbers for the action\_term category in the test set are not surprising as the number of trigger words in the dataset are far larger than the number of action term categories, which necessitated

construction of some rules for mapping. It is conceivable that the action\_term category is in fact better suited to a machine-learning approach than the rule-based approach that we used.

Entity class	Training dataset			Test dataset	
	Precision	Recall	Macro F1	Precision	Recall
Chemical	0.54	0.82	0.61	0.559	0.877
Disease	0.44	0.65	0.48	0.467	0.621
Gene	0.28	0.49	0.33	0.373	0.698
Action term	0.65	0.62	0.60	0.482	0.522

**Table 1.**

Overall, while our system performance in all categories clearly could do with substantial improvement, the scores shown above are somewhat similar to those reported in the CTD task in the BioCreative Workshop 2012 (1).

## References

1. Wiegers, T.C., Davis, A.P. and Mattingly, C.J. (2012). Collaborative biocuration - text-mining development task for document prioritization for curation. *Database (Oxford)*, 10.1093/database/bas037.

# A Web Service Annotation Framework for CTD Using the UIMA Concept Mapper

Andrew MacKinlay and Karin Verspoor

National ICT Australia, Victoria Research Laboratory  
Department of Computing and Information Systems, The University of Melbourne  
Melbourne 3010 VIC, Australia

## Introduction

We developed a set of simple web service annotators to address the named entity categories of *chemical*, *disease*, *gene*, and *action term* as defined by a set of controlled vocabularies in use at the Comparative Toxicogenomics Database [1]. It has been observed that when a large target vocabulary is defined, as is the case for gene normalization tasks and at least three of the tasks we tackle here, there may not be a significant advantage for methods that perform a separate mention detection step (e.g., using a classifier derived using machine learning) prior to mapping mentions to the target vocabulary, over strict dictionary-based methods [2]. This observation results from the requirement that in this scenario, all entity mentions must eventually be associated to a controlled vocabulary term, and we expect that association to be done by matching (possibly using “fuzzy” matching) mentions to the name or synonyms of the terms in the dictionary. Hence, we have implemented dictionary-based annotators.

## Dictionary methods

Our first suite of annotators is a dictionary-lookup system based on ConceptMapper<sup>1</sup> [3], a tool for finding dictionary matches in the UIMA framework [4, 5]. At initialisation, it is supplied with a prepared dictionary of terms. It keeps this potentially large collection of strings (containing single or multiple tokens) in memory in an efficient data structure and then searches text supplied to it for token sequences which match those in the dictionary. It has the ability to store synonyms for terms and map them back to a canonical form.

It has several parameters which control how the matching occurs, making the tool quite flexible. These include parameters relating to case matching, as well as flags for whether overlapping spans should all be annotated. It has been observed that the parameters and their values can significantly influence the performance of dictionary-based matching tools and appropriate values can vary with the specific type of entity to be recognized [6]. Therefore, some parameter settings are specific to particular target dictionary of the annotator, and are noted below. Of the parameters we set in all cases, the following are the most notable:

---

<sup>1</sup> <http://uima.apache.org/d/uima-addons-current/ConceptMapper/ConceptMapperAnnotatorUserGuide.html>

- `OrderIndependentLookup = false`; we only match if the order of the contained tokens is the same as those in the dictionary.
- `FindAllMatches = false`; we only find the longest match within a given span of text, ignoring any shorter spans contained in it.
- `SearchStrategy = ContiguousMatch`; we require that matched tokens be adjacent to each other.

It is also important that `ConceptMapper` is supplied with a high-quality dictionary, or the matches will be of poor quality. The dictionaries in all cases are derived from the appropriate provided CTD dictionaries, although the specifics of constructing them differ for each annotator, and are discussed in more detail below.

### **Dictionary-based Disease and Chemical Annotators**

While these annotators are applied to fairly different tasks, the techniques we settled on were identical for each. We created a `ConceptMapper` dictionary entry for each item in the corresponding CTD data set. We also created a synonym for each synonym listed in the supplied data, but excluded any synonyms with a length of one character or less, or which occurred in a list of common English words (derived from the Snowball project<sup>2</sup>). These heuristics for dictionary construction increased the precision over a subset of the learning corpus without harming recall. The `CaseMatch` flag of `ConceptMapper` was set to `ignoreall`, meaning that the dictionary match was case insensitive.

### **Dictionary-based Gene Annotator**

Using the same parameters as above for gene annotation led to a very high number of false positives, so in this case we added more restrictions during dictionary construction and annotation to increase precision. When creating the gene dictionary for `ConceptMapper`, as well as applying the same filter as for chemicals and diseases, we also exclude anything which is a sequence of 5 or fewer digits, anything which is a sequence of 1 or 2 known words, and anything which has fewer than 3 characters in total. We also change the `ConceptMapper CaseMatch` flag to `digitfold`, meaning that the case is ignored for entries containing one or more digits, but we require an exact match otherwise.

### **Dictionary-based Action Term Annotator**

The dictionary for action-term mapping was far smaller, and is also somewhat different in nature from the other annotators since derivational and inflectional variants of the source terms are far more likely to be important. We manually created a dictionary of morphological variants of the

---

<sup>2</sup> <http://snowball.tartarus.org/algorithms/english/stop.txt>

action terms based on those in the source CTD data, giving a dictionary of 273 synonyms in total. The `CaseMatch` flag was again set to `ignoreall`.

### **MetaMap-based methods**

We built an additional set of annotators for the *disease* and *chemical* categories, by taking advantage of the well-known MetaMap tool [7, 8]. We observed that the CTD dictionary for these two categories included MeSH (Medical Subject Headings) identifiers as an identifier or alternative identifier. Since the MetaMap tool includes the capability of recognizing MeSH terms, this seemed to be a logical strategy for recognizing these terms. We substituted our ConceptMapper-based annotators in our UIMA pipeline with the MetaMap UIMA Annotator,<sup>3</sup> limited MetaMap to recognizing terms from the MeSH vocabulary, and then matched the MeSH IDs to the appropriate CTD controlled vocabulary terms. As the performance of the MetaMap annotators was inferior to the dictionary-based lookup over a subset of the learning corpus, these were not submitted for the official evaluation.

### **Server implementation details**

We implemented each of our annotators as a service that calls a UIMA instance. Each annotator described above is associated with a UIMA pipeline. Each pipeline is instantiated once (initialisation of the ConceptMapper pipelines, in particular, is time-consuming, since all dictionary entries must be loaded into memory), and then called repeatedly from the web server as REST requests are received.

Our use of the UIMA framework meant that we only had to implement the web service connection to the underlying annotation system once; the communication between web service and a UIMA pipeline with a given type system is generic, while the internal characteristics of the UIMA pipeline itself can change. In addition, adapting to a new type system (for example the MetaMap type system rather than the types produced by ConceptMapper) is also straightforward.

### **Conclusions**

We were able to build simple annotators for the Comparative Toxigenomics Database by focusing on the dictionaries that define the target vocabulary of the database, and by building on existing tools for dictionary term matching – one generic, ConceptMapper, for which we only had to provide the appropriate dictionary specifications and select meaningful parameter settings, and one tailored to biomedical vocabulary, MetaMap, for which we had to subselect from its output the vocabulary relevant to the CTD categories. Our servers are intended to provide a simple annotation baseline for comparison with more sophisticated named entity recognition servers.

---

<sup>3</sup> [http://metamap.nlm.nih.gov/README\\_uima.html](http://metamap.nlm.nih.gov/README_uima.html)

## References

1. Davis AP, King BL, Mockus S, Murphy CG, Saraceni-Richards C, Rosenstein M, Wiegers T, Mattingly CJ: **The Comparative Toxicogenomics Database: update 2011.** *Nucleic Acids Research* 2011, **39**(suppl 1):D1067-1072.
2. Verspoor K, Roeder C, Johnson HL, Cohen KB, Baumgartner WA, Jr., Hunter LE: **Exploring species-based strategies for gene normalization.** *IEEE/ACM Trans Comput Biol Bioinform* 2010, **7**(3):462-471.
3. Tanenblatt MA, Coden A, Sominsky IL: **The ConceptMapper Approach to Named Entity Recognition.** In: *Proceedings of the NLP Frameworks Workshop at the Language Resources and Evaluation Conference.* 2010: 9-14.
4. Ferrucci D, and A. Lally: **UIMA: an architectural approach to unstructured information processing in the corporate research environment.** *Natural Language Engineering* 2004, **10**(3/4):327-348.
5. Ferrucci D, Lally A, Verspoor K, Nyberg E: **Unstructured Information Management Architecture (UIMA) Version 1.0.** In.: Oasis; 2009.
6. Funk C, Baumgartner Jr. W, Garcia B, Roeder C, Bada M, Cohen KB, Hunter LE, Verspoor K: **Large-scale biomedical concept recognition: An evaluation of current automatic annotators and their parameters.** *BMC Bioinformatics* to appear.
7. Aronson A, Lang F: **An overview of MetaMap: historical perspective and recent advances.** *Journal of the American Medical Informatics Association* 2010, **17**(3):229-236.
8. Aronson AR: **Effective Mapping of Biomedical Text to the UMLS Metathesaurus: The MetaMap Program.** In: *Proc AMIA 2001.* 2001: 17-21.

# Multi-stage Gene Mention Identification Method for BioCreative IV Track 3

Po-Ting, Lai<sup>1,2</sup>, Kuan-Chieh, Chung<sup>3</sup>, Johnny Chi-Yang Wu<sup>1</sup>, Hong-Jie Dai<sup>4,1</sup>, and Richard Tzong-Han Tsai<sup>5\*</sup>

<sup>1</sup>Institute of Information Science, Academia Sinica, Taipei, Taiwan, R.O.C., <sup>2</sup>Department of Computer Science, National Tsing-Hua University, HsinChu, Taiwan, R.O.C., <sup>3</sup>Department of Computer Science & Engineering, Yuan Ze University, Taoyuan, Taiwan, R.O.C., <sup>4</sup>Graduate Institute of BioMedical Informatics, College of Medical Science and Technology, Taipei Medical University, Taipei, Taiwan, R.O.C., <sup>5</sup>Department of Computer Science and Information Engineering, National Central University, Taoyuan, Taiwan, R.O.C.

## Abstract

In this paper, we describe our gene mention identification developed for the Comparative Toxicogenomic Database (CTD) track of BioCreative IV. We follow the configuration of the CTD track to implement our system, which can recognize gene mentions described in a given BioC document and return their official symbols. The system employs a multi-stage approach to identify gene mentions based on our gene normalization systems developed in the BioCreative II.5 interactor normalization task and BioCreative III gene normalization task. Through the participation of the track, we would like to see its effect and scalability in processing data within the context of the CTD curation process.

## Introduction

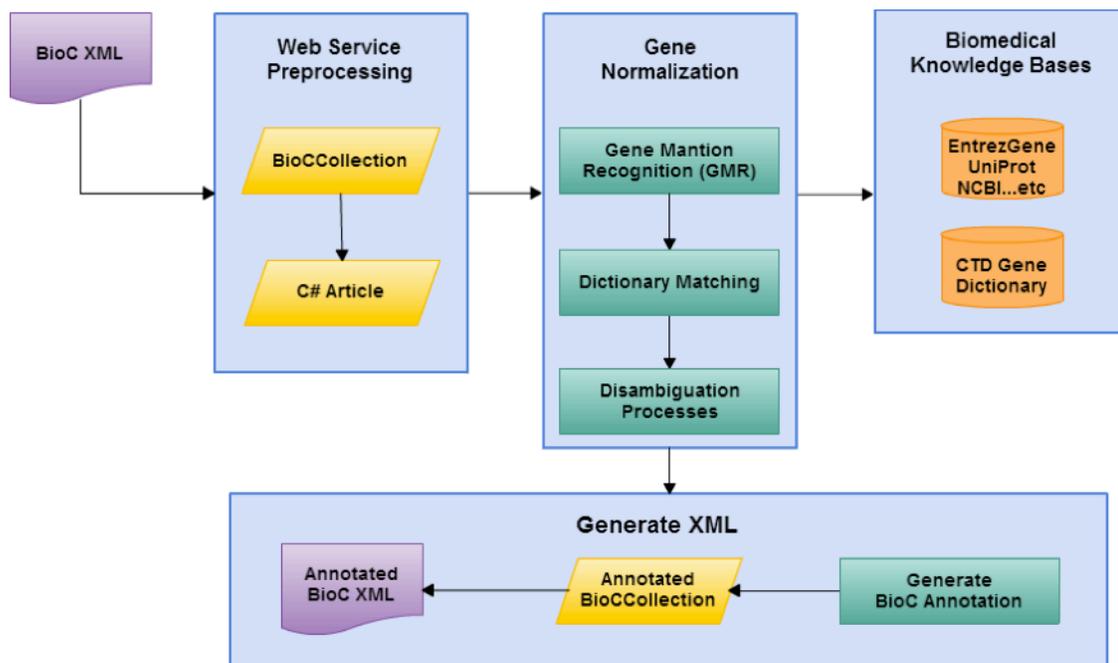
The goal of gene mention recognition (GMR) is to recognize gene mentions expressed in a narrative description. Several approaches have been proposed and achieved convincing performance in many international challenges and shared tasks [1, 2]. However, the results of GMR are still difficult to use directly because of the wide synonym and high ambiguity of variation in names across articles. Gene normalization (GN) goes beyond GMR by normalizing the gene mentions to a unique gene database ID [3]. By normalizing gene mentions to their IDs, we can determine their official symbol in attempt to reduce the ambiguity.

In the BioCreative II.5 interactor normalization task and BioCreative III GN task [4, 5], we have established a multi-stage GN approach and achieved satisfying results. Through the participation of the BioCreative IV Comparative Toxicogenomic Database (CTD) track, we would like to see its effect and scalability in processing data within the context of the CTD curation process. GN systems usually incorporate machine learning models for recognizing gene mentions, along with

---

<sup>1</sup>Corresponding authors

several biomedical knowledge bases to assign the proper ID to each gene mention. Our BioC-GN module incorporates a hybrid method to recognize gene mentions, and employs several rule-based disambiguation processes to select candidate gene IDs based on a multi-stage process. This developed BioC-GN module is available at [http://bws.iis.sinica.edu.tw/BioC\\_GN/RestServiceImpl.svc/gene](http://bws.iis.sinica.edu.tw/BioC_GN/RestServiceImpl.svc/gene).



**Figure 1:** BioC-GN module workflow for BioCreative IV CTD Track

## Method

### Service Architecture

Figure 1 depicts the service architecture of the developed BioC-GN module. The module allows clients to submit a PubMed abstract represented in the BioC XML format, and the server will provide an annotated XML in BioC format in return. After receiving client’s request, the XML is parsed into the Java object—BioCCollection. In order to apply our GN procedure, which is implemented in C# programming language, the BioCCollection object is translated into our C# Article object by using IKVM.NET Java virtual machine. Subsequently, we employ a dictionary-based tagger and our GMR model [6], which is trained on the BioCreative II dataset [7] with the CRF model, to recognize gene mentions. After recognizing gene mentions within the text, we apply orthogonal variation to map genes to their corresponding Entrez IDs. Furthermore, a disambiguation process is included to ensure the fidelity of normalization. The resulting unambiguous gene annotations are mapped to their relative gene symbol through the CTD gene

dictionary [8], and then transformed to BioC Annotation. Finally, the annotated BioCCollection will be returned to the client. We elaborate each step in the following subsections.

### **Gene Mention Recognition**

GMR recognizes gene mentions within plain text, and it is usually a machine learning technique dependent process. For this procedure, we employ two GMR taggers. The first is a dictionary-based tagger that uses the CTD gene vocabulary to match gene mentions. The second is a machine learning-based tagger in which we follow the IOB2 format [9] and use Conditional Random Fields model [10] with several features such as morphological features, syntactic features, and collocation features [6]. Morphological features are used in regard to the morphological properties shared by gene names. For instance, "HLA-A" and "HLA-B" will be normalized to the same term "AAA\_A", and "IL-2" and "IL-21" will be normalized to the same term "A\_1". Syntactic features including part-of-speech and Chunk features are used to identify NE boundaries. For example, verbs and prepositions usually indicate the boundary of an NE. In addition to the previous unigram features, collocation features are exploited as bigram and trigram word features. For instance, when the previous word is "transcription," and the current word is "factor," the current word is most likely the last word of a transcription factor name, which is categorized as a protein name in the GENIA ontology. If the two taggers generate overlapped gene boundaries, we select the one with larger boundary, while the other is saved for the disambiguation process; if the one with larger boundary cannot be mapped/disambiguated, the stored name with shorter boundary will be reassessed.

### **Dictionary Matching**

Dictionary Matching is able to assign candidate identifiers to each recognized gene mention. We use a dictionary compiled by the CTD gene dictionary and generate their orthographical variants. The dictionary is further expanded by collecting gene names in the EntrezGene and UniProt database. Each recognized gene mention is looked up in the dictionary by using two matching methods, exact matching and partial matching, developed in our previous work [11].

### **Multistage-disambiguation Process**

If a gene mention is mapped to two or more gene identifiers in the dictionary matching step, the disambiguation process determines which is more appropriate. In our previous work [11], we have constructed several GN rules, which utilize context information such as chromosome location, sequence length, and so on to determine the given identifier's label. For example, the rule

$$hasChromosomeInfo(id, s) \wedge hasCandidate(x, id) \Rightarrow NormalizeTo(x, id)$$

indicates that if the chromosome location information of the gene mention  $x$ , which has the  $id$  as its candidate ID, can be found in the surrounding context  $s$ ,  $x$  should be linked to  $id$ . Each rule is

associated with a weight, and the final disambiguation step is based on the linear combination of the weighted scores of the various rules.

After the disambiguation step, names of the successfully mapped gene mentions and their corresponding database IDs are collected to generate a refinement dictionary. Refinement is then performed by using the exact matching algorithm to search the whole article for mentions in this dictionary that were not recognized by GMR. If the refinement process assigns a gene mention boundary overlapped with existing recognized gene mentions and their IDs are different, the same disambiguation process is re-conducted in case the new candidate ID has not been considered in the previous judgment.

## Reference

1. J.-D. Kim, T. Ohta, Y. Tsuruoka, and Y. Tateisi, "Introduction to the bio-entity recognition task at JNLPBA," *Proceedings of the International Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA-04)*, pp. 70-75, 2004.
2. A. Yeh, A. Morgan, M. Colosimo, and L. Hirschman, "BioCreative IV task 1A: gene mention finding evaluation," *BMC Bioinformatics*, vol. 6, p. S2, 2005.
3. A. A. Morgan, Z. Lu, X. Wang, A. M. Cohen, J. Fluck, P. Ruch, A. Divoli, K. Fundel, R. Leaman, J. Hakenberg, C. Sun, H.-h. Liu, R. Torres, M. Krauthammer, W. W. Lau, H. Liu, C.-N. Hsu, M. Schuemie, K. B. Cohen, and L. Hirschman, "Overview of BioCreative II gene normalization," *Genome Biology*, vol. 9, p. S3, 2008.
4. F. Leitner, S. A. Mardis, M. Krallinger, G. Cesareni, L. A. Hirschman, and A. Valencia, "An Overview of BioCreative II.5," *IEEE/ACM TRANSACTIONS ON COMPUTATIONAL BIOLOGY AND BIOINFORMATICS*, vol. 7, pp. 385-399, 2010.
5. Z. Lu, H.-Y. Kao, C.-H. Wei, M. Huang, J. Liu, C.-J. Kuo, C.-N. Hsu, R. T.-H. Tsai, H.-J. Dai, N. Okazaki, H.-C. Cho, M. Gerner, I. Solt, S. Agarwal, F. Liu, D. Vishnyakova, P. Ruch, M. Romacker, F. Rinaldi, S. Bhattacharya, P. Srinivasan, H. Liu, M. Torii, S. Matos, D. Campos, K. Verspoor, K. M. Livingston, and W. J. Wilbur, "The gene normalization task in BioCreative III," *BMC Bioinformatics*, vol. 12, p. S2, 2011.
6. R. Tsai, C. Sung, H. Dai, H. Hung, T. Sung, and W. Hsu, "NERBio: using selected word conjunctions, term normalization, and global patterns to improve biomedical named entity recognition," *BMC Bioinformatics*, vol. 7 Suppl 5, p. S11, 2006.
7. L. Smith, L. K. Tanabe, R. J. n. Ando, C.-J. Kuo, I.-F. Chung, C.-N. Hsu, Y.-S. Lin, R. Klinger, C. M. Friedrich, K. Ganchev, M. Torii, H. Liu, B. Haddow, C. A. Struble, R. J. Povinelli, A. Vlachos, W. A. B. Jr, L. Hunter, B. Carpenter, R. T.-H. Tsai, H.-J. Dai, F. Liu, Y. Chen, C. Sun, S. Katrenko, P. Adriaans, C. Blaschke, R. Torres, M. Neves, P. Nakov, A. Divoli, M. Maña-López, J. Mata, and W. J. Wilbur, "Overview of BioCreative II gene mention recognition," *Genome Biology*, vol. 9, p. S2, 2008.
8. M. D. I. B. Laboratory. (August 14). *CTD gene dictionary*. Available: <http://ctdbase.org/downloads/#allgenes>

9. E. F. T. K. Sang and J. Veenstra, "Representing text chunks," presented at the Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics, Bergen, Norway, 1999.
10. A. McCallum, "Efficiently inducing features of conditional random fields," presented at the Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence, Acapulco, Mexico, 2003.
11. H.-J. Dai, P.-T. Lai, and R. T.-H. Tsai, "Multistage Gene Normalization and SVM-Based Ranking for Protein Interactor Extraction in Full-Text Articles," *IEEE/ACM TRANSACTIONS ON COMPUTATIONAL BIOLOGY AND BIOINFORMATICS*, vol. 7, pp. 412-420, 2010.

# ToxiCat: Hybrid Named Entity Recognition services to support curation of the Comparative Toxicogenomic Database

Dina Vishnyakova<sup>1,2,4,\*</sup>, Julien Gobeill<sup>1,3,4</sup>, Emilie Pasche<sup>1,2,3,4</sup> and Patrick Ruch<sup>1,3,4</sup>

<sup>1</sup> Bibliomics and Text Mining (BiTeM) Group: <http://bitem.hesge.ch>

<sup>2</sup> Division of Medical Information Sciences, University and University Hospitals of Geneva

<sup>3</sup> Information Science Department, HES-SO/University of Applied Science Geneva

<sup>4</sup> SIBtex, SIB Swiss Institute of Bioinformatics, Geneva, Switzerland.

\*Corresponding author: SIMED; University Hospitals of Geneva; 4, rue Gabrielle-Perret-Gentil; CH-1211 Geneva 14; Tel: +41 22 372 61 99; email: [dina.vishnyakova@hcuge.ch](mailto:dina.vishnyakova@hcuge.ch)

## Abstract

We report on the original implementation of named entity recognition (NER) modules based on an automatic text categorization pipeline, so-called ToxiCat (Toxicogenomic Categorizer), developed to perform biomedical documents classification and prioritization for the previous Biocreative campaign in order to speed up the curation of the Comparative Toxicogenomics Database (CTD). ToxiCat NER modules are a group of components that analyse text for enclosed information. These modules are based on an information retrieval engine for MEDLINE (EAGLi), a gene normalization (GN) service (NormaGene) developed for a previous BioCreative campaign, gene ontology categorizer (GOCat) and finally an entity recognizer for diseases and chemicals. The NER services are publically available as RESTful web services at <http://pingu.unige.ch:8080/Toxicat>.

## Introduction

The recognition of biomedical concepts in texts is a key technology for automatic or semi-automatic analysis of textual resources. Most of applications are based on Named Entity Recognition (NER) tools in information retrieval, information extraction and document classification tasks. In recent years, NER systems development has reached great attention in the bioinformatics community. Multiple systems and algorithms have been developed and implemented. These systems and algorithms can be roughly split into 3 categories: rule-based and dictionary-based systems, fully automatic machine-learning systems and hybrids approaches, combining first two categories. Most tools require the user to specify certain configuration settings, like choosing a dictionary or creating an appropriate corpus of annotated texts in order to perform a reliable assessment where the operation to find or to design such a dataset could be time-consuming.

The work we present here is focused on the construction of some NER tools for the curation of the CTD (1), where the main accents are put on the identification of gene/protein, chemical, disease, and chemical/gene-specific action term mentions, each within the context of CTD's controlled vocabulary structure. We should notice that there are several information available about the development of the NER systems for gene/protein, chemical or disease concepts while the identification of a chemical/gene-specific action term is covered only within the framework of CTD. The representation of a chemical/gene-specific action term in a text is often not implicit. We have used components such as EAGLi's Keyword extractor (2) and NormaGene (3) in order to ease the process of systems configuration or to avoid time-consuming dataset-couple processes and finally GOCat (4) to solve the problem with the chemical/gene-specific action term recognition.

## Data and Methods

### Data overview

The CTD track of BioCreative IV proposes to focus on the interoperability, e.g. to explore how text-mining methods can successfully be applied to practically help biocuration of a large molecular biology knowledge base. The main objective of the Track-3 task is to provide Web Services for concepts annotations to maintain the Comparative Toxicogenomics Database (CTD) with the interacting entities (small molecules and gene products) and the pathologies likely to reflect the toxicity of the chemical compound.

The organizers provided a learning corpus in BioC XML format of 1,112 abstracts for training; all curated gene/protein, chemical, and disease actors, and associated chemical/gene-specific action terms. Each curated interaction associated with the article is also provided. Testing set consisted of 510 documents.

### Methods

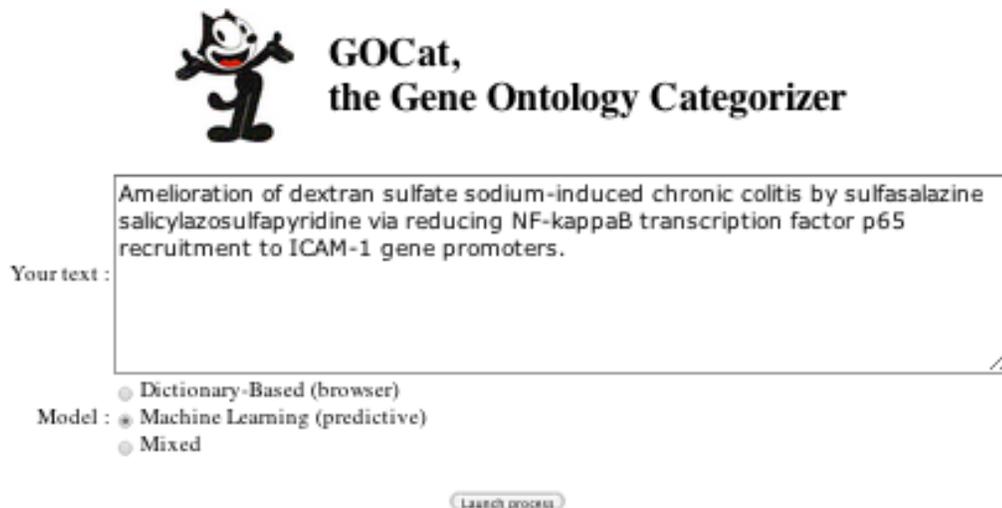
We designed NER services for each category of interest, i.g. for gene/protein, disease, chemical and chemical/gene action term as following:

- **Gene/Protein NER:** We based our NER service for gene/protein concepts on the NormaGene named-entity normalizer (3,5). This gene and protein named-entity recognizer was developed for the BioCreative III task to address the gene normalization task (3). Like other named-entity recognizer, it identifies the patterns of the gene and protein name as well as it attempts to assign a unique identifier. Thus, NormaGene also attempts to recognize, when possible, what organisms is mentioned in the text to link properly a gene/protein name with a unique sequence. Internally, NormaGene is able to recognize all gene candidates stored in the Gene and Protein Synonyms DataBase (GPSDB) (6), as well as all species stored in NEWT ([www.ebi.ac.uk/newt/](http://www.ebi.ac.uk/newt/))), which is appropriate to annotate contents for UniProt/SwissProtKB but which does exceed the

coverage of CTD (7). The internal dictionaries of NormaGene are therefore reduced to curate CTD. Finally, results returned by NormaGene are compared to the CTD genes controlled vocabulary to further reduce the list of results. The controlled vocabulary of CTD contains over 257.000 NCBI genes' identifiers and over 479.000 genes' names including synonyms (5). If the entities recognized by NormaGene are found in the CTD genes' vocabulary then we extract all synonyms based on the approved genes ID and match them against the abstract. Indeed, gene and protein identifiers suggested by NormaGene cannot always be explicitly found in the body of the input document as NormaGene uses a generative model, which exploits also functional similarities (3) and not only textual similarities. Additionally, as a final check all candidates selected by NormaGene NER tool are matched against synonyms from a provided dictionary of CTD.

- **Disease/Chemical NERs:** We created an ad-hoc keyword recognizer for diseases and chemicals. This keyword recognizer is based on the controlled vocabularies provided by CTD. Unlike the previous results of CTD Triage task in Biocreative 2012, where systems showed high results based on Recall the current task (Track-3-CTD) is taking into account the Precision of the system, see (5) for more details. Disease/Chemical NER relies on the UMLS Metathesaurus. For both chemical and disease entities, a Word-Sense Disambiguator (WSD) is created, based on the UMLS Semantic Types (5, 8). In (5) we have described in details which types of chemicals and diseases were eliminated from the final result. Further, in order to avoid common English words in the list of candidates, we created a common English word recognizer based on a general-purpose English corpora. Unspecific disease and chemical names were thus discarded.
- **Chemical/Gene action term NER:** CTD curates specific chemical–gene and protein interactions in vertebrates and invertebrates from the published literature. Most interactions are binary, involving one chemical and one gene or protein. After exploring the corpus provided by the organizers we found that the information about chemical/gene action term is not represented explicitly in the text of the provided corpus. Since the concept of action term identification in the text is not widely covered by the bioinformatics community, it makes the task especially complex. We assumed that to use Gene Ontology could help to identify action terms. Ontological approaches rely on formal ontological principles to formalize the relations expected between biological entities according to general theories specified in some upper-level ontologies (9). In the Gene Ontology, we can observe that several chemical entities are found in GO descriptors and synonyms (9). Consequently we attempted to assign some GO concepts to the input text using the Gene Ontology Categorizer – GOCat (4), see Fig.1 and Fig 2. GOCat is a state-of-the-art thesaurus-based system combined with a machine learning system (4). The output of GOCat is a ranked list of candidate GO terms, which are the most likely to characterize the functional profile of a given abstract. Next, we process GOCat results with the developed NER based on a dictionary where all action terms

provided by organizers (<http://ctdbase.org/help/ixnQueryHelp.jsp#actionType>) were included.



**Figure 1.** The GOCat interface where as an input the user can provide a text, e.g. an abstract of the document and choose the processing of results between three models: Dictionary-Based, Machine learning and Mixed.



**Figure 2.** An example of the GOCat output returned for the input of “Amelioration of dextran sulfate sodium-induced chronic colitis by sulfasalazine salicylazosulfapyridine via reducing NF-kappaB transcription factor p65 recruitment to ICAM-1 gene promoters.” Here, the output is a list of the most associated GO concepts, which are split into the three GO axes: molecular functions, biological processes and cellular components.

## Results and Conclusion

The results of ToxiCat (Group 183), computed on the official data provided by BioCreative 2012's organizers using official metrics, are shown in Table 1, where:

- Curated Actors - the terms curated for the current document by CTD in the respective NER category.
- Text Mined Actors - the text mined terms returned from the NER Web Service on a provided document
- Text Mined Actors Hits - provides an explanation of how matches between the curated terms and the text mined terms were determined.

**Table1.** Toxicat NER services results of the Track-3.

	<b>disease</b>	<b>chemical</b>	<b>gene/protein</b>	<b>action term</b>
<b>Records Processed</b>	510	510	510	510
<b>Text Mined Actors</b>	795	1156	1062	1763
<b>Text Mined Actor Hits</b>	366	685	370	450
<b>Curated Actors</b>	943	1192	1122	966
<b>Micro-Average Recall Aggregate Curated Actors</b>	0.388	0.57	0.32	0.46
<b>Macro-Average Recall Aggregate Curated Actors</b>	0.396	0.56	0.35	0.45
<b>Micro-Average Precision</b>	0.46	0.59	0.348	0.255
<b>Macro-Average Precision</b>	0.40	0.55	0.342	0.259
<b>Average Seconds Processed</b>	0.43	0.83	4.40	24.22

In BioCreative IV, Track 3 was investigated to interoperability and efficiency aspects; therefore the ability to integrate a particular workflow and the processing time were assessed. In Table 1, disease and chemical NER services show quite acceptable average response time compared to gene and action term NER services. This result suggests that a general-purpose gene normalizer and the NER service based on gene ontology categorizer are time-consuming for a specific database curation task. However, action term NER service is competitive regarding the recall compare to disease and gene/proteins NERs. At the same time such text processing tools (NormaGene and GOCat) can particularly address situations where training data are not available.

On the training data, our NER services obtained a precision of 77% for chemicals and 72% for disease and a recall 74% and 69% respectively. Then, we applied these settings to the official data. The results in Table 1 showed some overfitting phenomena, e.g entities detected by NER were rejected by the WSD from the final results.

Although current results seem suggesting that text mining can effectively help curators' tasks by providing access to more relevant contents, it is worth noticing that the effectiveness of some NERs can be obtained by specializing some of the general-purpose text mining tools. Finally, we plan to further investigate text-mining tools, which can be integrated into a biocuration process and can decrease time-consuming factor in situations where training data are not available.

## References

1. Wieggers T., (2012) Collaborative Biocuration-Text Mining Development Task for Document Prioritization for Curation. Proceedings of BioCreative 2012.
2. Ruch P. (2006) Automatic assignment of biomedical categories: toward a generic approach. *Bioinformatics*. **22**(6):658-64
3. Lu Z., Wilbur W J., et al. (2011) The gene normalization task in BioCreative III. *BMC Bioinformatics*, **12**(Suppl 8):S2
4. Gobeill J, Pasche E., Vishnyakova D. and Ruch P (2013), Managing the data deluge: data-driven GO category assignment improves while complexity of functional annotation increases. *Database*, doi: 10.1093/database/bat041
5. Vishnyakova D, Pasche E, Ruch P. (2012) Using binary classification to prioritize and curate articles for the Comparative Toxicogenomics Database. *Database (Oxford)*.;2012:bas050.
6. Pillet V, Zehnder M, Seewald AK, Veuthey AL, Petrak J. (2005) GPSDB A new database for synonyms expansion of gene and protein names. *Bioinformatics*. **21**(8):1743-4.
7. Davis AP, Wieggers TC, Murphy CG, and Mattingly CJ. (2011). The curation paradigm and application tool used for manual curation of the scientific literature at the Comparative Toxicogenomics Database. *Database*, Oxford
8. Jimeno-Yepes A., McInnes B., Aronson A. (2011) Collocation analysis for UMLS knowledge-based word sense disambiguation. *BMC Bioinformatics*, **12**(Suppl 3):S4
9. Burgun, A., & Bodenreider, O. (2005, April). An ontology of chemical entities helps identify dependence relations among Gene Ontology terms. *In First Symposium on Semantic Mining in Biomedicine*.

# Adapting the OCMiner text processing system to the CTD controlled vocabulary

Matthias Irmer<sup>1</sup>, Claudia Bobach<sup>1</sup>, Timo Böhme<sup>1</sup>, Ulf Laube<sup>1</sup>, Anett Püschel<sup>1</sup>, Lutz Weber<sup>1,\*</sup>

<sup>1</sup>OntoChem GmbH, Halle (Saale), Germany

\*Corresponding author: E-mail: lutz.weber@ontochem.com

## Abstract

We adapted OCMiner, a modular text processing pipeline especially suited for high-speed processing of large document collections, to a specific controlled vocabulary as given by the Comparative Toxicogenomic Database (CTD). We provide a RESTful web service which processes documents given in the BioCreative XML format and annotates them with domain-specific terms from the CTD domains genes, chemistry, diseases and action terms.

**Keywords:** Text mining; Named entity recognition; Controlled vocabulary; Domain-specific annotation

## Introduction

The Comparative Toxicogenomic Database (CTD) is a publicly available resource consisting of accumulated knowledge in the domains chemistry, genes/proteins, and diseases. The database contains both manually curated and automatically inferred chemical-gene/protein interactions, chemical-disease relationships, and gene-disease relationships. The BioCreative CTD task consisted in providing a web service for automatically annotating documents with domain-specific terms contained in the CTD database.

We adapted OCMiner, a modular text annotation and information extraction system especially suited for high-speed processing of large document collections, to the specific controlled vocabulary of CTD terms.

## System description

### Preparatory work

In a first step, the given CTD ontologies / taxonomies were converted into the OBO format (OBO foundry, <http://www.obofoundry.org/>) according to OntoChem's standard procedure. Synonyms were generated, cleaned and expanded:

- synonyms were generated from the name,
- synonyms with comma were transformed into comma-free synonyms with a term reversal around the comma: e.g. “Calculus, Kidney” into “Kidney Calculus”,
- duplicates were removed,
- OBO attributes were added to each synonym (synonym scope and type e.g. EXACT SYNONYM, as well as source, date, language and prelabel). The prelabel synonym is typically the originally supplied name.

Domain-specific blacklists, whitelists and graylists were created. They include stop words, common words and the like. Especially in the case of proteins/genes, there is a considerable amount of synonyms which are homonyms to common English words (e.g. "the", "and"). In order to get meaningful annotation results, these terms have to be blacklisted. Some terms can be identified as gene/protein synonyms in a particular context only (e.g. enumerations). These terms are collected in a "graylist" and are only annotated under specific circumstances. Furthermore, there are conditional black- and whitelists where context conditions can be specified. For instance, the term "localization" is not annotated as an action term when preceded by "histochemical".

### **Processing pipeline**

For the BioCreative CTD annotation task, we make use of our OCMiner text mining system. It is a high-throughput UIMA-based (<http://uima.apache.org/>) modular framework designed for the rapid annotation of huge collections of texts of various categories (abstracts, journal articles, patents, technical documentations) and formats (PDF, XML, HTML,...). The overall architecture is depicted in Fig. 1.

**Figure 1.** OCMiner UIMA pipeline

The *collection reader* reads text data from varying sources (here: XML documents via web service). Then, a number of preparatory modules make sure that the text becomes processable. First of all, XML tags are separated from the proper text. Then, whitespaces and special characters are normalized, before the text is tokenized and ready for Named Entity recognition.

The most important components in the pipeline are the dictionary-based *domain annotators*. From the converted taxonomies, a dictionary was created for each domain. The domain annotators use these dictionaries, which are designed for high-speed lookup of terms in texts. A special feature is the ability to deal with typical variations in domain term usage. For instance, the protein term "5-HT2A" may be written as "5HT2A" or "5HT-2A".

Additional components handle specific scenarios. For instance, we use an *abbreviation annotator* which finds expansions of acronyms and abbreviated terms. The *coordinated entity annotator* recognizes expressions like "vitamine A and B" as a coordinated entity and annotates "vitamine A" as such and "B" as "vitamine B".

In a *post-processing* step, annotations are validated and cleaned in a configurable manner. As a general rule, we do not allow overlapping annotations. If an annotated term (e.g. "protease") is subsumed by another annotated term (possibly from another domain, e.g. "protease inhibitor"), then only the longer term is kept. Similarly, graylisted terms become available as annotations if they are part of enumerations of terms from the same domain.

Finally, a *consumer* writes the annotated data in a specific format to files, databases or indexes. For the BioCreative CTD task, the original XML file is augmented with annotations at a specified position and sent as a response to the web service query.

Annotated domain	Precision (macro-averaged)	Recall (macro-averaged)	Average seconds processed
gene	0.4454	0.6727	0.1419
disease	0.4583	0.5944	0.1404
chem	0.6244	0.8540	0.1407
action term	0.3996	0.3635	0.1396

**Table 1.** Final challenge results for CTD annotation

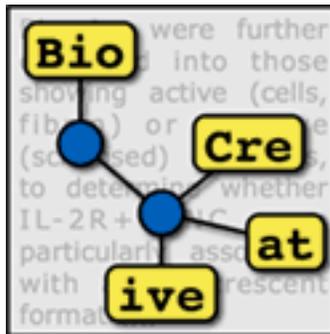
## Results

In the BioCreative CTD challenge, we obtained the final results given in Table 1. Note that the retrieval of action terms did not coincide very much with the manually curated information. This

is due to the fact that in many cases curators tagged a document with a specific "action term" expressing a relationship chemistry/genes, genes/diseases, or chemistry/diseases, while the relationship was not explicitly expressed in the text using these terms. This suggests that a "pure" named entity recognition as applied here might not be a suitable method for relationship extraction. On the other hand, results in the other domains were much better, with best results in the chemistry domain.

## **Funding**

This work was supported by a grant of the German ministry of education and research (BMBF), project "SARminer" [grant number 01IS12011A].



## TRACK 4 (GO)

### Organizers:

- Zhiyong Lu, National Center for Biotechnology Information (NCBI), NIH, USA
- Donghui Li, TAIR
- Cecilia N. Arighi, University of Delaware
- Kimberly Van Auken, WormBase, USA

# The Gene Ontology Task at BioCreative IV

Yuqing Mao<sup>1</sup>, Kimberly Van Auken<sup>2</sup>, Donghui Li<sup>3</sup>, Cecilia N. Arighi<sup>4</sup>, Zhiyong Lu<sup>1,\*</sup>

<sup>1</sup>National Center for Biotechnology Information (NCBI), 8600 Rockville Pike, MD 20817 USA

<sup>2</sup>WormBase, Division of Biology, California Institute of Technology, 1200 E. California

Boulevard, Pasadena, CA 91125 USA <sup>3</sup>Department of Plant Biology, The Arabidopsis

Information Resource, Carnegie Institution for Science, Stanford, CA 94305, USA <sup>4</sup>Center for Bioinformatics and Computational Biology, University of Delaware, Newark, DE 19711, USA

\*Corresponding author: Tel: 301 594 7089, E-mail: Zhiyong.Lu@nih.gov

## Abstract

Gene Ontology (GO) annotation is a common task among model organism database (MOD) groups. It is a very time-consuming and labor-intensive task, thus often considered as one of the bottlenecks in literature curation. There is a growing need for semi- or fully-automated GO curation techniques that will help database curators rapidly and accurately identify gene function information in full-length articles. Despite multiple attempts in the past, few studies have proven to be useful with regard to assisting real-world GO curation. The lack of relevant training data and opportunities for interaction between text mining developers and GO curators has limited the advances in algorithm development and corresponding use in practical circumstances. To this end, we organized a text-mining challenge task for literature-based GO annotation in BioCreative IV. More specifically, we developed two sub-tasks: a) to automatically locate text passages that contain GO-relevant information (a text retrieval task) and b) to automatically identify relevant GO terms for the genes in a given article (a concept recognition task). With the support from five MODs, we provided teams with nearly 4,000 unique text passages that served as the basis for each GO annotation in our task data. Such evidence text information has long been recognized as critical for text-mining algorithm development but was never made available due to the high cost of curation. In total, seven teams participated in the challenge task. From the team results, we find an overall improvement in performance for recognizing GO terms when comparing to similar task results in the past. Future work should focus on improving performance of GO concept recognition and incorporating practical benefits of text-mining tools into real-world GO annotation.

## Introduction

Manual Gene Ontology (GO) annotation is the task of human curators assigning gene functional information using GO terms through reading the biomedical literature. This is a common task among Model Organism Database (MOD) groups (1) and can be time-consuming and labor-intensive. Thus, manual GO annotation is often considered one of the bottlenecks in literature-

based biocuration (2). As a result, many MODs can only afford to curate a fraction of relevant articles. For instance, the curation team of The Arabidopsis Information Resource (TAIR) has been able to curate less than 30% of newly published articles that contain information about Arabidopsis genes (3).

Recently, there is a growing interest for building automatic text-mining tools to assist manual biocuration (4-10), including systems that aim to help database curators rapidly and accurately identify gene function information in full-length articles (11,12). Although automatically mining GO terms from full-text articles is not a new problem in BioNLP, few studies have proven to be useful with regard to assisting real-world GO curation. The lack of access to relevant evidence text associated with GO annotations and limited opportunities for interaction with actual GO curators have been recognized as the major difficulties in algorithm development and corresponding application in practical circumstances (12,13). As such, in BioCreative IV, not only do we plan to provide teams with article-level gold-standard GO annotations for each full-text article as has been done in the past, but we will also provide evidence text for each GO annotation with the help from expert GO curators. That is, to best help text-mining tool advancement, evidence text passages that support each GO annotation will be provided in addition to the usual GO annotations which typically include three distinct elements: gene or gene product, GO term, and GO evidence code.

Also as we know from past BioCreative tasks, recognizing gene names and experimental codes from full text are difficult tasks on their own (14-17). Hence, to encourage teams to focus on GO term extraction, we proposed, for this task, to separate gene recognition from GO term and evidence code selection by including both the gene names and associated NCBI Gene identifiers in the task data sets.

Specifically, we propose two challenge tasks towards automated GO concept recognition from full-length articles:

#### **Task A: Retrieving GO evidence text for relevant genes**

GO evidence text is critical for human curators to make associated GO annotations. For a given GO annotation, multiple evidence passages may appear in the paper, some being more specific with experimental information while others may be more succinct about the gene function. For this sub-task, participants are given as input full-text articles together with relevant gene information. For system output, teams have to submit a list of GO evidence sentences for each of the input genes in the paper. Manually curated GO evidence passages will be used as the gold standard for evaluating team submissions. Each team is allowed to submit 3 runs.

## **Task B: Predicting GO terms for relevant genes**

This sub-task is a step towards the ultimate goal of using computers for assisting human GO curation. As in Task A, participants are given as input full text articles with relevant gene information. For system output, teams are asked to return a list of relevant GO terms for each of the input genes in a paper. Manually curated GO annotations will be used as the gold standard for evaluating team predictions. Similar to Task A, each team is allowed to submit 3 runs.

Generally speaking, the first sub-task is a text retrieval task while the second can be seen as a multi-class text classification problem where each GO term represents a distinct class label. In the BioNLP research domain, the first sub-task is similar to the BioCreative I GO sub-task 2.1 (12), BioCreative II Interaction Sentence sub-task (14), and automatic GeneRIF identification (18-20). The second sub-task is similar to the BioCreative I GO sub-task 2.2 (12) and is also closely related to the problem of semantic indexing of biomedical literature such as automatic indexing of biomedical publications with MeSH terms (21-24).

## **Methods**

### **Corpus annotation**

A total of 8 professional GO curators from five different MODs (FlyBase; MaizeGDB; RGD; TAIR; WormBase) contributed to the development of the task data. To create the annotated corpus, each curator was asked, in addition to their routine annotation of gene-related GO information, to mark up the associated evidence text in each paper that supports those annotations using a Web-based annotation tool. To provide complete data for text-mining system development (i.e., both positive and negative training data), curators were asked to select evidence text exhaustively throughout the paper (25).

For obtaining high-quality and consistent annotations across curators, detailed annotation guidelines were developed and provided to the curators. In addition, each curator was asked to practice on a test document following the guidelines before they begin curating task documents. Due to the significant workload and limited number of curators per group, each paper was only annotated by a single curator.

### **Evaluation measures**

For Task A evaluation, traditional precision (P), recall (R) and  $F_1$  score ( $F_1$ ) are reported when comparing the submitted gene-specific sentence list against the gold standard. We computed the numbers of true positives (TP) and false positives (FP) in two ways: the first one (exact match) is a strict measure that requires the returned sentences exactly match the sentence boundary of human markups while the second (overlap) is a more relaxed measure where a prediction is considered correct (i.e. TP) as long as the submitted sentence overlaps with the gold standard.

$$P = \frac{tp}{tp + fp}, R = \frac{tp}{tp + fn}, F_1 = 2 \cdot \frac{P \times R}{P + R}$$

For the Task B evaluation, gene-specific GO annotations in the submissions will be compared with the gold standard. In addition to the traditional precision, recall and F1 score, hierarchical Precision (hP), Recall (hR) and F-score (hF<sub>1</sub>) will also be computed where common ancestors in both the computer-predicted and human-annotated GO terms are considered. The second set of measures were proposed to reflect the hierarchical nature of GO: a gene annotated with one GO term is implicitly annotated with all of the term’s parents, up to the root concept (26,27). Such a measure takes into account that “predictions that are close to the oracle label should score better than predictions that are in an unrelated part of the hierarchy.” (26) Specifically, the hierarchical measures are computed as:

$$hP = \frac{\sum_i |\hat{G}_i \cap \hat{G}'_i|}{\sum_i |\hat{G}'_i|}, hR = \frac{\sum_i |\hat{G}_i \cap \hat{G}'_i|}{\sum_i |\hat{G}_i|}, hF_1 = 2 \cdot \frac{hP \cdot hR}{hP + hR}$$

$$\hat{G}_i = \{\bigcup_{G_k \in G_i} \text{Ancestors}(G_k)\}$$

$$\hat{G}'_i = \{\bigcup_{G'_k \in G'_i} \text{Ancestors}(G'_k)\}$$

where  $\hat{G}_i$  and  $\hat{G}'_i$  are the respective sets of ancestors of the computer-predicted and human-annotated GO terms for the *i*th gene.

## Results

### The BC4GO corpus

The task participants were provided with three data sets comprising a total of 200 full-text articles in the BioC XML format (28). Our evaluation for the two sub-tasks was to respectively assess teams’ ability to return relevant sentences and GO terms for each given gene in the 50 test articles. Hence, we show in Table 1 the overall statistics of the BC4GO corpus including the numbers of genes, gene-associated GO terms and evidence text passages. For instance, in the 50 test articles, 194 genes were associated with 644 GO Terms, and 1,681 evidence text passages, respectively. We refer interested readers to (25) for a detailed description of the BC4GO corpus.

**Table 1.** Overall statistics of the BC4GO corpus.

Curated Data	Training Set	Dev. Set	Test Set
<b>Full text articles</b>	100	50	50
<b>Genes</b> in those articles	300	171	194
Gene-associated <b>passages</b> in those articles	2,234	1,247	1,681
Gene-associated <b>GO terms</b> in those articles	954	575	644

### Team participation results

Overall, seven teams (3 from Americas, 3 from Asia, and 1 from Europe) participated in the GO task. In total, they submitted 32 runs: 15 runs from five different teams for Task A, and 17 runs from six teams for Task B.

### Team Results of Task A

Table 2 shows the results of 15 runs submitted by the five participating teams in Task A. Run 3 from Team 238 achieved the highest  $F_1$  score in both exact match (0.270) and overlap (0.387) calculations. Team 238 is also the only team who submitted results for all 194 genes from the input of the test set. The highest recall is 0.424 in exact match and 0.716 in overlap calculations by the same run (Team 264, run 1), respectively. The highest precision is 0.220 in exact match by Team 238 Run 2 and 0.354 in overlap by Team 183, Run 2.

**Table 2.** Team results for Task A using traditional Precision (P), Recall (R) and F-Mesure (F1). Both strict exact match and relaxed overlap measure are considered.

Team	Run	Genes	Passages	Exact match			Overlap		
				P	R	F <sub>1</sub>	P	R	F <sub>1</sub>
183	1	173	1,042	0.206	0.128	0.158	0.344	0.213	0.263
183	2	173	1,042	0.217	0.134	0.166	<b>0.354</b>	0.220	0.271
183	3	173	1,042	0.107	0.066	0.082	0.204	0.127	0.156
237	1	23	54	0.185	0.006	0.012	0.333	0.011	0.021
237	2	96	2,755	0.103	0.171	0.129	0.214	0.351	0.266
237	3	171	3,717	0.138	0.305	0.190	0.213	0.471	0.293
238	1	194	2,698	0.219	0.352	<b>0.270</b>	0.313	0.503	0.386
238	2	194	2,362	<b>0.220</b>	0.310	0.257	0.314	0.442	0.367
238	3	194	2,866	0.214	0.366	<b>0.270</b>	0.307	0.524	<b>0.387</b>
250	1	161	3,297	0.146	0.286	0.193	0.239	0.469	0.317
250	2	140	2,848	0.153	0.259	0.193	0.258	0.437	0.325
250	3	161	3,733	0.140	0.311	0.193	0.226	0.503	0.312
264	1	167	13,533	0.052	<b>0.424</b>	0.093	0.088	<b>0.716</b>	0.157
264	2	111	2,243	0.037	0.049	0.042	0.077	0.103	0.088
264	3	111	2,241	0.037	0.049	0.042	0.077	0.103	0.088

### Team Results of Task B

Table 3 shows the results of 17 runs submitted by the six participating teams in Task B. Run 1 from Team 183 achieved the highest  $F_1$  score in traditional (0.134) and hierarchical measures (0.338). The same run also obtained the highest precision of 0.117 in exact match while the highest precision in hierarchical match is 0.415 obtained by the Run 1 of Team 237. However, note that this run only returned 37 GO terms for 23 genes. The highest recall is 0.306 and 0.647 in the two measures by Run 3 of Team 183.

**Table 3.** Team results for the Task B using traditional Precision (P), Recall (R) and F1-measure (F1) and hierarchical precision (hP), recall (hR) and F1-measure (hF1).

Team	Run	Genes	GO terms	Exact match			Hierarchical match		
				P	R	F <sub>1</sub>	hP	hR	hF <sub>1</sub>
183	1	172	860	<b>0.117</b>	0.157	<b>0.134</b>	0.322	0.356	<b>0.338</b>
183	2	172	1720	0.092	0.245	<b>0.134</b>	0.247	0.513	0.334
183	3	172	3440	0.057	<b>0.306</b>	0.096	0.178	<b>0.647</b>	0.280
220	1	50	2639	0.018	0.075	0.029	0.064	0.190	0.096
220	2	46	1747	0.024	0.065	0.035	0.087	0.158	0.112
237	1	23	37	0.108	0.006	0.012	<b>0.415</b>	0.020	0.039
237	2	96	2424	0.108	0.068	0.029	0.084	0.336	0.135
237	3	171	4631	0.037	0.264	0.064	0.150	0.588	0.240
238	1	194	1792	0.054	0.149	0.079	0.243	0.459	0.318
238	2	194	555	0.088	0.076	0.082	0.250	0.263	0.256
238	3	194	850	0.029	0.039	0.033	0.196	0.310	0.240
243	1	109	510	0.073	0.057	0.064	0.249	0.269	0.259
243	2	104	393	0.084	0.051	0.064	0.280	0.248	0.263
243	3	144	2538	0.030	0.116	0.047	0.130	0.477	0.204
250	1	171	1389	0.052	0.112	0.071	0.174	0.328	0.227
250	2	166	1893	0.049	0.143	0.073	0.128	0.374	0.191
250	3	132	453	0.095	0.067	0.078	0.284	0.161	0.206

## Discussion and Conclusions

As mentioned earlier, our task is related to a few previous challenge tasks on biomedical text retrieval and semantic indexing. In particular, our task resembles the earlier GO task in BioCreative I (12). On the other hand, our two sub-tasks are also different from the previous tasks. For the passage retrieval task, we only provide teams with pairs of <gene, document> and ask their systems to return relevant evidence text while <gene, document, GO terms> triples were provided in the earlier task.

For the GO term prediction task, we provided teams with the same <gene, document> pairs and asked their systems to return relevant GO terms. In addition to such input pairs, the expected number of GO terms and their associated GO branches (molecular function, biological process, and cellular component) returned were also provided in the earlier task. Another difference is that along with each predicted GO term for the given gene in the given document, output of associated evidence text is also required in the earlier task.

Finally, the evaluation mechanism differed in the two challenge events. We provided the reference data prior to the team submission and preformed standard evaluation. By contrast, in the BioCreative I GO task, no gold-standard evaluation data were provided before the team

submission. Instead, expert GO curators were asked to manually judge the team submitted results. Such a post-hoc analysis could miss true positives not returned by teams and would not permit evaluation of new systems after the challenge.

In summary, we provided less input information to teams in both sub-tasks and followed protocols of standard challenge evaluation – two major differences between our task and the previous BioCreative I task (12). This is partly because we aim to have our tasks resemble real-world GO annotation more closely, where the only input to human curators is the set of documents. Despite these differences, we were intrigued by any potential improvement in the task results due to the advancement of text mining research in recent years. Since the ultimate goal of the task is to find GO terms, the results of Task B are of more interest and significance in this aspect, though evidence sentences are of course important for reaching this goal. By comparing the team results in the two challenge events (Table 3 above vs. Table 5 in (12)), we can observe a general trend of performance increase on this task over time. For example, the best-performing team in 2005 (12) was only able to predict 78 TPs (out of 1227 in gold standard) – a recall of less than 7% – while there are several teams in our task who obtained recall values between 10% and 30%. The numbers are even greater when measured by taking account of the hierarchical nature of the Gene Ontology.

Despite these encouraging results, overall team results suggest that automatically mining GO terms from literature remains very challenging due to difficulties in multiple aspects: First, the number of GO terms (class labels for classification) is extremely large: there are over 40,000 unique GO concepts to date. Second, GO terms (and associated synonyms) are designed for unifying gene function annotations rather than for text mining, and are therefore rarely found verbatim in the article. For example, our analysis shows that only about 1/3 of the annotated GO terms in our corpus can be found using exact matches in their corresponding articles. On the other hand, not every match related to a GO concept is annotated. Instead, only those GO terms that represent experimental findings in a given full-text paper are selected. Hence, automatic methods must be able to filter irrelevant mentions that share names with GO terms (e.g. the GO term ‘growth’ is a common word in articles, but additional contextual information would be required to determine if this relatively high-level term should be used for GO annotation purposes). Finally, human annotation data for building statistical/machine-learning approaches is still lacking. Despite our best efforts, we are only able to include 200 annotated articles in our corpus, which contains evidence text for only 1,311 GO terms.

Our challenge task was inspired and developed in response to the actual needs of GO manual annotation. However, compared to the real-world GO annotation, the BioCreative challenge task is simplified in two aspects: a) gene information is provided to the teams while in reality they are unknown; and b) extraction of GO evidence code information is not required for our task while it is an essential part of the GO annotations in practice. Further investigation of automatic

extraction of gene and evidence code information, along with corresponding GO terms, remains as future work.

## Acknowledgments

We are grateful to the support of our corpus curators. We also thank Lynette Hirschman, John Wilbur, Cathy Wu, Kevin Cohen, Martin Krallinger, and Thomas Wieggers from the BioCreative IV organizing committee for their support, and Judith Blake, Andrew Chatr-aryamontri, Sherri Matis, Fiona McCarthy, Sandra Orchard, and Phoebe Roberts from the BioCreative IV User Advisory Group for their helpful discussions. This research is supported by NIH Intramural Research Program, National Library of Medicine (YM & ZL).

## References

1. Balakrishnan, R., Harris, M.A., Huntley, R., *et al.* (2013) A guide to best practices for Gene Ontology (GO) manual annotation. *Database : the journal of biological databases and curation*, **2013**, bat054.
2. Lu, Z., Hirschman, L. (2012) Biocuration workflows and text mining: overview of the BioCreative 2012 Workshop Track II. *Database : the journal of biological databases and curation*, **2012**, bas043.
3. Li, D., Berardini, T.Z., Muller, R.J., *et al.* (2012) Building an efficient curation workflow for the Arabidopsis literature corpus. *Database : the journal of biological databases and curation*, **2012**, bas047.
4. Wu, C.H., Arighi, C.N., Cohen, K.B., *et al.* (2012) BioCreative-2012 virtual issue. *Database : the journal of biological databases and curation*, **2012**, bas049.
5. Arighi, C.N., Carterette, B., Cohen, K.B., *et al.* (2013) An overview of the BioCreative 2012 Workshop Track III: interactive text mining task. *Database : the journal of biological databases and curation*, **2013**, bas056.
6. Wei, C.H., Harris, B.R., Li, D., *et al.* (2012) Accelerating literature curation with text-mining tools: a case study of using PubTator to curate genes in PubMed abstracts. *Database : the journal of biological databases and curation*, **2012**, bas041.
7. Neveol, A., Wilbur, W.J., Lu, Z. (2012) Improving links between literature and biological data with text mining: a case study with GEO, PDB and MEDLINE. *Database : the journal of biological databases and curation*, **2012**, bas026.
8. Wei, C.-H., Kao, H.-Y., Lu, Z. (2012) PubTator: A PubMed-like interactive curation system for document triage and literature curation. *Proceedings of the BioCreative 2012 workshop*, Washington, D.C., pp. 20-24.
9. Wei, C.-H., Kao, H.-Y., Lu, Z. (2013) PubTator: a Web-based text mining tool for assisting Biocuration. *Nucleic Acids Res*, **41**, W518-W522.
10. Neveol, A., Wilbur, W.J., Lu, Z. (2011) Extraction of data deposition statements from the literature: a method for automatically tracking research results. *Bioinformatics*, **27**, 3306-3312.
11. Van Auken, K., Jaffery, J., Chan, J., *et al.* (2009) Semi-automated curation of protein subcellular localization: a text mining-based approach to Gene Ontology (GO) Cellular Component curation. *BMC Bioinformatics*, **10**, 228.

12. Blaschke, C., Leon, E.A., Krallinger, M., *et al.* (2005) Evaluation of BioCreative assessment of task 2. *BMC Bioinformatics*, **6 Suppl 1**, S16.
13. Camon, E.B., Barrell, D.G., Dimmer, E.C., *et al.* (2005) An evaluation of GO annotation retrieval for BioCreative and GOA. *BMC Bioinformatics*, **6 Suppl 1**, S17.
14. Krallinger, M., Leitner, F., Rodriguez-Penagos, C., *et al.* (2008) Overview of the protein-protein interaction annotation extraction task of BioCreative II. *Genome Biol*, **9 Suppl 2**, S4.
15. Lu, Z., Kao, H.Y., Wei, C.H., *et al.* (2011) The gene normalization task in BioCreative III. *BMC Bioinformatics*, **12 Suppl 8**, S2.
16. Lu, Z., Wilbur, W.J. (2010) Overview of BioCreative III Gene Normalization. *Proceedings of the BioCreative III workshop*, Bethesda, USA, pp. 24-45.
17. Van Landeghem, S., Bjorne, J., Wei, C.H., *et al.* (2013) Large-scale event extraction from literature with multi-level gene normalization. *PLoS One*, **8**, e55814.
18. Cohen, A.M., Hersh, W.R. (2006) The TREC 2004 genomics track categorization task: classifying full text biomedical documents. *J Biomed Discov Collab*, **1**, 4.
19. Lu, Z., Cohen, K.B., Hunter, L. (2006) Finding GeneRIFs via gene ontology annotations. *Pac Symp Biocomput*, 52-63.
20. Lu, Z. (2007) Text Mining on GeneRIFs. *Computational Bioscience Program*. University of Colorado School of Medicine, Aurora, USA, Vol. Ph.D. thesis.
21. Huang, M., Neveol, A., Lu, Z. (2011) Recommending MeSH terms for annotating biomedical articles. *J Am Med Inform Assoc*, **18**, 660-667.
22. Neveol, A., Shooshan, S.E., Humphrey, S.M., *et al.* (2009) A recent advance in the automatic indexing of the biomedical literature. *Journal of biomedical informatics*, **42**, 814-823.
23. Vasuki, V., Cohen, T. (2010) Reflective random indexing for semi-automatic indexing of the biomedical literature. *Journal of biomedical informatics*, **43**, 694-700.
24. Huang, M., Lu, Z. (2010) Learning to annotate scientific publications. *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*. Association for Computational Linguistics, Beijing, China, pp. 463-471.
25. Auken, K.V., Schaeffer, M.L., McQuilton, P., *et al.* (2013) Corpus Construction for the BioCreative IV GO Task. *Proceedings of the BioCreative IV workshop*, Bethesda, USA.
26. Eisner, R., Poulin, B., Szafron, D., *et al.* (2005) Improving protein function prediction using the hierarchical structure of the Gene Ontology. *Proceedings of 2005 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*.
27. Kiritchenko, S., Matwin, S., Famili, A.F. (2005) Functional annotation of genes using hierarchical text categorization. *Proceedings of the BioLINK SIG: Linking Literature, Information and Knowledge for Biology*.
28. Comeau, D.C., Islamaj Dogan, R., Ciccarese, P., *et al.* (2013) BioC: a minimalist approach to interoperability for biomedical text processing. *Database : the journal of biological databases and curation*, **2013**, bat064.

# Corpus Construction for the BioCreative IV GO Task

Kimberly Van Auken<sup>1</sup>, Mary L. Schaeffer<sup>2</sup>, Peter McQuilton<sup>3</sup>, Stanley J. F. Lauderkind<sup>4</sup>, Donghui Li<sup>5</sup>, Shur-Jen Wang<sup>4</sup>, G. Thomas Hayman<sup>4</sup>, Susan Tweedie<sup>3</sup>, Cecilia N. Arighi<sup>6</sup>, James Done<sup>1</sup>, Hans-Michael Müller<sup>1</sup>, Paul W. Sternberg<sup>1,7</sup>, Yuqing Mao<sup>8</sup>, Chih-Hsuan Wei<sup>8</sup> and Zhiyong Lu<sup>8,\*</sup>

<sup>1</sup>Division of Biology, California Institute of Technology, 1200 E. California Blvd., Pasadena, CA 91125, USA

<sup>2</sup>USDA-ARS Plant Genetics Research Unit and Division of Plant Sciences, Department of Agronomy, University of Missouri, Columbia, MO 65211, USA

<sup>3</sup>FlyBase, Department of Genetics, University of Cambridge, Downing Street, Cambridge CB2 3EH, UK

<sup>4</sup>Rat Genome Database, Human and Molecular Genetics Center, Medical College of Wisconsin, 8701 Watertown Plank Road, Milwaukee, WI 53226, USA

<sup>5</sup>Department of Plant Biology, Carnegie Institution for Science, 260 Panama Street, Stanford, CA 94305, USA

<sup>6</sup>Center for Bioinformatics and Computational Biology, University of Delaware, Newark, DE 19711, USA

<sup>7</sup>Howard Hughes Medical Institute, California Institute of Technology, 1200 E. California Blvd., Pasadena, CA 91125, USA

<sup>8</sup>National Center for Biotechnology Information, 8600 Rockville Pike, Bethesda, MD 20894, USA

\*Corresponding author. Tel: 301-594-7089 Email: zhiyong.lu@nih.gov

## Abstract

Gene function curation via Gene Ontology (GO) annotation is a common task among Model Organism Database (MOD) groups. Due to its manual nature, this task is time-consuming and labor-intensive, and thus considered one of the bottlenecks in literature curation. There have been many previous attempts of automatic identification of GO terms and associated information from full text. However, few systems have delivered an accuracy that is comparable to human annotators. One recognized challenge in developing such systems is the lack of marked passage-level evidence text that provides the basis for making GO annotations. To this end, we aim to create a corpus that includes the GO evidence text along with the three essential elements of GO annotations: 1) a gene or gene product, 2) a GO term and 3) a GO evidence code. To ensure our results are consistent with real-life GO annotation data, we recruited a team of eight professional GO curators from the biocuration community, and asked them to follow their routine GO annotation protocols. With the aid of a web-based annotation tool, our annotators marked up

nearly 4,000 unique text passages in 200 full-text articles where on average each unique GO term is annotated with four different evidence text passages. Further, our corpus analysis shows that most of the evidence text occurs in the body of the article while only as little as 12% appears in the abstracts. This result demonstrates the necessity of text mining of full text for finding GO terms. Through its use as the official data set for the BioCreative IV GO (BC4GO) task, we expect our unique BC4GO corpus to become a valuable resource for the BioNLP research community.

## **Introduction**

The Gene Ontology (GO) (<http://www.geneontology.org>) is a controlled vocabulary for standardizing the description of gene and gene product attributes across species and databases (1). Currently, there are about 40,000 GO terms that are organized in a hierarchical manner under three GO sub-categories: molecular function, biological process and cellular component. Since its inception, GO terms have been used in over 126 million annotations to over 9 million gene products (2). The accumulated GO annotations have been shown to be increasingly important in an array of different areas of biological research such as high-throughput omics data analysis and the study of developmental biology (3-5).

Among the 126 million GO annotations, most are derived from automated techniques such as mapping of GO terms to protein domains, motifs (InterPro2GO) (6) or corresponding concepts in one of the controlled vocabularies by UniProt (7); only a very small portion (1.1 million) are derived from manual curation of published experimental results in the biomedical literature (8). While the former approach is efficient in assigning higher-level GO terms, the latter provides more reliable and detailed GO annotations that are critical for the kinds of analyses mentioned above. Generally speaking, the manual GO annotation process first involves the retrieval of relevant publications. Once found, the full text is manually inspected to identify the gene product of interest, the relevant GO terms, and the evidence code to indicate the type of supporting evidence, e.g. mutant phenotype or genetic interaction, for inferring the relationship between a gene product and a GO term. Such a process is time-consuming and labor intensive, and thus many MODs are confronted with a daunting backlog of GO annotation. For instance, in recent years, TAIR's curation team has been able to curate only a fraction of newly published articles that contain information about Arabidopsis genes (<30%) (9). It is thus clear that the manual curation process requires computer assistance, and this is seen in a growing interest in, and need for semi- or fully automated curation pipelines for assisting biocuration (10-20). In particular, a number of studies (21-29) have attempted to (semi-)automatically predict GO terms from text including a previous BioCreative challenge task (30). However, few studies have proven to be useful with regard to assisting real world GO curation. Based on a recent study, enhanced text-mining capabilities to automatically recognize GO terms from full text remains one of the most in-demand tasks among the biocuration community (31).

As concluded in the previous BioCreative task (30,32), one of the main difficulties was “the lack of a high quality training set consisting in the annotation of relevant text passages”. Such a training set in practice provides the evidence for human curators to make associated GO annotations. To advance the development of automatic systems for GO curation, we propose to create a corpus that includes the GO evidence text along with three essential elements of GO annotations: 1) a gene or gene product, 2) a GO term (e.g., receptor-mediated endocytosis), and 3) a GO evidence code (e.g., Inferred from Mutant Phenotype (IMP)). The evidence texts for GO annotations may be derived from a single sentence, or multiple continuous, or discontinuous, sentences. The evidence for a GO annotation could also be derived from multiple lines of experimentation, leading to multiple text passages in a paper supporting the same annotation. Since many learning-based text-mining algorithms rely on both positive and negative training instances, it is important to be as thorough as possible when manually annotating sentences. It is therefore important to capture all of the curation-relevant sentences to ensure the positive and negative sets are as distinct as possible.

The exhaustive capture of evidence text in full-length articles makes our dataset, namely the BC4GO corpus, unique among the many previously annotated corpora (e.g.(33-36)) for the BioNLP research community. To our best knowledge, BC4GO is the only publicly available corpus that contains textual annotation of GO terms in accordance with the general practice of GO annotation (8) by professional GO curators. For instance, while in a previous study (17) every mention related to a GO concept was annotated, in BC4GO we have annotated only those GO terms that represent experimental findings in a given full-text paper.

## **Methods and Materials**

### **Annotators**

Through the BioCreative IV User Advisory Group, we recruited eight expert curators from five different MODs: FlyBase (2 curators), MaizeGDB (1 curator), RGD (3 curators), TAIR (1 curator), and WormBase (1 curator). All our curators are experienced in GO manual annotation.

### **Annotation Guidelines**

For achieving consistent annotations between annotators, the task organizers followed the usual practice of corpus annotation (33-37): first we drafted a set of annotation guidelines and then asked each of our annotators to practice them on a shared article as part of the training process. The results of their annotations on the common article were shared among all annotators and subsequently the discrepancies in their annotations were discussed. Based on the discussion, the annotation guidelines were revised accordingly. For brevity, we only discuss below the two kinds of evidence text passages we chose to capture. The detailed guidelines are publicly available at the corpus download website: <http://www.biocreative.org/resources/corpora/bc-iv-go-task-corpus/>

**1. Experiment Type:** These sentences describe experimental results and can be used to make a complete GO annotation (i.e., the entity being annotated, GO term, and GO evidence code). The annotation of such sentences is required throughout the paper, including the abstract, and any supporting summary paragraphs such as ‘Author summary’ or ‘Conclusions’.

*Ex1: On the other hand, the amount of UNC-60B-GFP was reduced and UNC-60A-type mRNAs, UNC60A-RFP and UNC-60A-Experiment, were detected in asd-2 and sup-12 mutants (Figure 2H, lanes 2 and 3), consistent with their colour phenotypes shown in Figure 2C and 2A, respectively. (PMC3469465)*

This sentence contains information about:

The gene/protein entities: *asd-2* and *sup-12*

GO term: regulation of alternative mRNA splicing, via spliceosome (GO:0000381)

GO evidence code: Inferred from Mutant Phenotype (IMP)

**2. Summary Type:** Distinct from statements that describe the details of experimental findings, papers also include many statements that summarize these findings. These summary statements don’t necessarily indicate exactly *how* the information was discovered, but often contain concise language about *what* was discovered. Such sentences are helpful to capture because they may inform GO term selection in a concise manner despite the lack of information about evidence code selection.

*Ex2: Taken together, our results demonstrate that muscle-specific splicing factors ASD-2 and SUP-12 cooperatively promote muscle-specific processing of the unc-60 gene, and provide insight into the mechanisms of complex pre-mRNA processing; combinatorial regulation of a single splice site by two tissue-specific splicing regulators determines the binary fate of the entire transcript. (PMC3469465)*

The gene/protein entities: ASD-2 and SUP-12

GO term: regulation of alternative mRNA splicing, via spliceosome (GO:0000381)

GO evidence code: N/A

### Article Selection

The 200 articles in the BC4GO corpus are chosen from annotators’ existing annotation workload at their respective MODs. Such a protocol minimizes the additional workload to our curators while at the same time guarantees the curated papers are representative of real-life GO annotations. Another requirement is that annotated articles are published in a list of select journals (e.g. PLoS Genetics) in PubMed Central (PMC) that allow free access and text analysis.

## Annotation Tool

A web-based annotation tool was developed for use in the annotation process as shown below in Figure 1. The tool allows the upload of full text articles in either HTML or XML formats and subsequently displays the article in a Web browser. Currently, the tool allows the annotator to select and highlight a single sentence, or multiple sentences (regardless of whether they are contiguous or not) as GO evidence text. When a sentence is highlighted, a pop-up window appears for annotators to enter required GO annotation information: a GO term, a GO evidence code, and associated gene(s). The tool also allows the annotators to preview their annotations before committing them to the database. Annotation results of each paper can be downloaded as HTML files.

Figure 5  
Overexpression of the nlp-29 locus.  
The GATA transcription factor ELT-3 fulfils a generic requirement for nlp-29 expression

Inspection of the upstream sequences of genes of the nlp-29 cluster revealed the presence of a conserved putative GATA site in the promoter regions of nlp-28 to nlp-31 (Figure S6). The GATA factor ELT-2 has been shown to be important for the control of infection-inducible gene expression in the intestine [26]. There are 14 GATA factors encoded in the *C. elegans* genome [27]. We focused on those known to be expressed in the epidermis or seam cells, namely elt-1, 3 and 6 and egl-18 (previously known as elt-5) [28]–[30]. RNAi of egl-18, elt-1 and 6 did not have a significant effect (results not shown). We observed, however, that the constitutive expression of pmlp-29::GFP and its induction by infection or high salt was reduced upon elt-3 RNAi. We confirmed this effect using an elt-3 null mutant allele and found that GFP expression was knocked down by half following either of these treatments, as well as in untreated worms. The level of red fluorescence, from the pcol-12::DsRed transgene was, on the other hand, essentially the same (+/-15%) in all cases (Figure 6A). To assay for a role of elt-3 in fungal resistance, we compared the survival of wild-type and mutant worms after *D. coniospora* infection. Unlike the nlp-29(tm1931) mutant, which behaved essentially like the wild type, there was a marked reduction in the resistance of the elt-3 mutants. These mutants, however, had a substantially reduced lifespan in the absence of infection. The same phenotypes were observed for tir-1(tm3036) mutants (Figure 6F & 6G). Thus, while being suggestive, we cannot definitively assign a specific role in fungal resistance to elt-3.

Figure 6  
Figure 6  
The GATA factor ELT-3 is required for gene induction in the epidermis.

Exposure to high salt up-regulates expression of the pgdph-1::GFP reporter. Unlike pmlp-29::GFP (Figure 6B & 6D). Interestingly, in the elt-3 mutant background, an abrogation of the epidermal e (6E). This suggests that although elt-3 is necessary for expression of AMP genes, it acts as a gene

### Discussion

#### Transcriptional response of *C. elegans* to fungal infection

In this study, after an unbiased microarray analysis of genes affected by natural fungal infection in a class of up-regulated genes. Synthetic NLP-31 has demonstrated antimicrobial activity in vitro against therefore candidate AMPs. Our sequence analysis showed that these proteins can be differentiated NLP-34 (but not NLP-32) carry the name Neuropeptide-Like Protein only for historical reasons. They possess antimicrobial activities [15], expression and biochemical analyses are needed to test if the

A very recent study reported changes in host gene expression induced by the nematode-trapping fungus with *M. haplotylum* used microarrays with probes to only a few hundred *C. elegans* genes, and of & S1C). Nevertheless, several nlp genes, including nlp-29, as well as cnc-4, were found to be induced that colonize the nematode intestine [14],[22],[26], another recent report indicates that infection of a pathogen infects worms via the uterus. A second Gram-positive bacterium, *M. nematophilum*, and indeed any of the nlp or cnc genes [33]. On the other hand, wounding the epidermis also provokes signalling pathway [19]. So both the nature of the pathogen and the route of infection likely play a

Link parameters  
textpresso-dev.caltech.edu/gsa/GO/popup.html

**name or sentence:**  
We observed, however, that the constitutive expression of pmlp-29::GFP and its induction by infection or high salt wa

Show Annotation Write Annotation Link! Clear

URL:  
GO Term (1):  
GO Evidence Code (1): with  
Gene (1):  
Comment (1):

**Figure 1.** Screenshot of the annotation tool. When a line or more of text is highlighted, a pop-up window appears where annotation data is entered.

## Final Data Dissemination

Both full-text articles and associated GO annotations (downloaded from PMC and the annotation tool, respectively) were further processed before releasing to the task participants. Specifically, we chose to format our data using the recently developed BioC standard for improved interoperability (38). First, for the 200 full-text articles, we converted their XMLs from the PMC format to the BioC format. Next, we extracted annotated sentences from downloaded HTML files and identified their offsets in the generated BioC XML files. Finally, for each article we created a corresponding BioC XML file for the associated GO annotations. Figure 2 shows a snapshot of our final released annotation files where one complete GO annotation is presented with the BioC format. For the gene entity, we provide both the gene mention as it appeared in the text and its corresponding NCBI Gene identifier.

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE collection SYSTEM "BioC.dtd">
<collection>
  <source>GO_Annotation</source>
  <date>20130316</date>
  <key>go_annotation.key</key>
  <document>
    <id>23840682</id>
    <passage>
      <infony key="type">abstract</infony>
      <offset>89</offset>
      <annotation id="23840682_1">
        <infony key="gene">emb16(100170235)</infony>
        <infony key="go-term">embryo development|GO:0009790</infony>
        <infony key="goevidence">IMP</infony>
        <infony key="type">GOA</infony>
        <location offset="415" length="114"/>
        <text>The emb16 mutation arrests embryogenesis at transition stage and allows the
          endosperm to develop largely normally.</text>
      </annotation>
    </passage>
  </document>
</collection>

```

**Figure 2.** A sample of GO annotation in BioC format.

## Results and Discussion

### Corpus Statistics

The task participants are provided with three data datasets comprising a total of 200 full-text articles. Table 1 shows the number of articles curated by each MOD. On average, each curator contributed about 25 articles for the task during this time period.

**Table 1.** Number of curated articles per MOD.

Data Set	FlyBase	MaizeGDB	RGD	TAIR	WormBase	Total
Training Set	19	21	43	10	7	100
Development Set	8	5	25	4	8	50
Test Set	12	4	20	7	7	50
Subtotal per team	39	30	88	21	22	200

Table 2 shows the main characteristics of the BC4GO corpus. Each annotation includes four elements: the gene/protein entity, GO term, GO evidence code, and evidence text (See **Figure** ). Note that one text passage can often provide evidence for annotating more than one gene, as well as more than one GO term. Therefore, we show in the last column of Table 2 the counts of evidence text passages in three different ways. The first number shows that the total number of text passages with respect to GO annotations: Over 5,000 text passages were used in the annotation of 1,311 unique GO terms. So on average, each GO term is associated with four different evidence text passages in our corpus. The second number (5,162) shows the total number of text passages with respect to different genes: For each of the 665 unique genes in our corpus, there are about 7.8 associated text passages. Finally, the last number is the total number of unique text passages annotated in our corpus regardless of their association to either gene or GO terms.

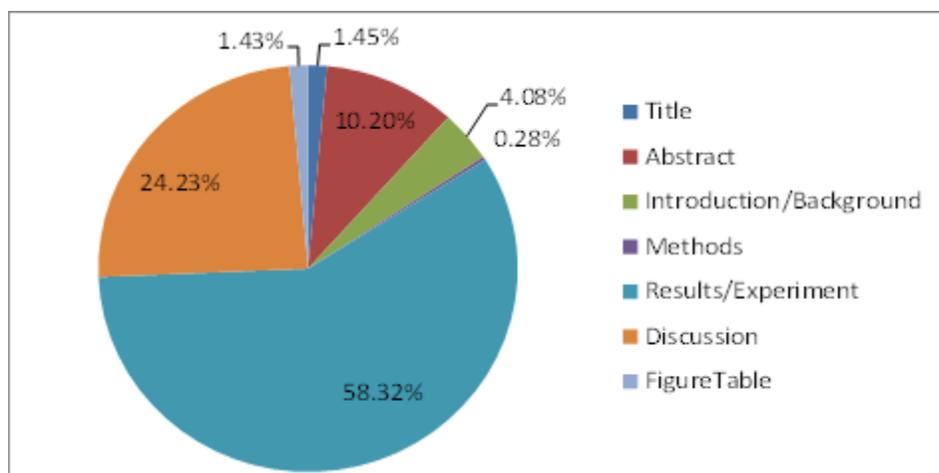
**Table 2.** Overall statistics of the annotated corpus.

Data Set	Articles	Genes (unique)	GO terms (unique)	Evidence text passages w.r.t. GO/Gene/Unique
Training Set	100	300	566	2,213/2,234/1,704
Development Set	50	171	367	1,299/1,247/963
Test Set	50	194	378	1,763/1,681/1,253
Total	200	665	1,311	5,275/5,162/3,920

From Table 2, we can compute that the average number of genes annotated in each article is 3.3, and the average number GO terms associated with each gene is 2.0 in our corpus. Furthermore, as mentioned before, we have annotated two types of evidence text, depending on whether they contain experimental information or not. Accordingly, the two kinds are distinguished in our annotations by the presence or absence of associated evidence code. For the total 3,920 unique pieces of evidence text, the majority (~70%) of them contain experimental evidence.

### The location of evidence text in the paper

Figure 3 shows the proportion of all evidence text in different parts of the article. As can be seen, the most informative location for extracting GO evidence text is the Results section, followed by the Discussion Section. Some GO evidence text also appears in the Table or Figure legend. Within the full text article, the Introduction/Background and Methods sections contain the least amount of information for complete GO annotation. Figure 3 also shows the limitation of using article abstracts for GO annotation: only 11.65% of the annotated text is found in the Title and Abstract combined. This finding further confirms the importance of using full text for GO annotation.



**Figure 3.** The proportion of annotated evidence text in different parts of the article.

## **Conclusions and Future Work**

Through collaboration with professional GO curators from five different MODs, we created a corpus for the development and evaluation of automated methods for identifying GO terms from full-text articles. The resulting BC4GO corpus is large-scale and the only one of its kind. We expect our BC4GO corpus to become a valuable resource for the BioNLP research community. We hope to see improved performance and accuracy of text mining for GO terms through the use of our annotated corpus in the BioCreative IV GO task and beyond.

There are several limitations of this work that warrant further investigation. First, in order to ensure the positive and negative sentences are as distinct as possible, we asked our annotators to mark up every occurrence of GO evidence text. As a result, it greatly increased the annotation workload for each individual annotator. Meanwhile, to maximize the number of annotated articles, we chose to assign one annotator per article. In other words, our articles are not double annotated. Second, despite all our best efforts in ensuring consistent annotations (e.g. creating annotation guidelines, and providing annotator training), there will always be variation in the depth of annotation between curators and organisms. For instance, there may be gray areas where some curators will select a sentence relating to a phenotype as a GO sentence, while others do not. In the future, we plan to assess the inter-annotator agreement for our corpus.

## **Authors' Contributions:**

Conceived and designed the annotation experiment: ZL, KVA, DL, CNA. Developed the annotation guidelines: ZL, KVA, PM, DL, ST. Developed the annotation tool: JD, KVA, HMM, PWS. Performed the annotation experiment: MLS, PM, SJFL, KVA, DL, SJW, GTH, ST, CNA. Analyzed the annotated data: YM, CW, ZL. Wrote the paper: ZL. All authors read and approved the final manuscript.

## **Acknowledgments**

We would like to thank Don Comeau, Rezarta Dogan and John Wilbur for general discussion and technical assistance in using BioC, and in particular to Don Comeau for providing us source PMC articles in the BioC XML format. We also thank Lynette Hirschman, Cathy Wu, Kevin Cohen, Martin Krallinger, and Thomas Wiegers from the BioCreative IV organizing committee for their support, and Judith Blake, Andrew Chatr-aryamontri, Sherri Matis, Fiona McCarthy, Sandra Orchard, and Phoebe Roberts from the BioCreative IV User Advisory Group for their helpful discussions. This research is supported by the Intramural Research Program of the NIH, National Library of Medicine (CW, YM, & ZL), by the USDA ARS (MLS), by grants from the National Human Genome Research Institute at the US National Institutes of Health # HG004090, # HG002223 and # HG002273, and by National Science Foundation grant ABI-1062520.

## References

1. Harris, M.A., Clark, J., Ireland, A., *et al.* (2004) The Gene Ontology (GO) database and informatics resource. *Nucleic Acids Res*, **32**, D258-261.
2. Balakrishnan, R., Harris, M.A., Huntley, R., *et al.* (2013) A guide to best practices for Gene Ontology (GO) manual annotation. *Database : the journal of biological databases and curation*, **2013**, bat054.
3. Hill, D.P., Berardini, T.Z., Howe, D.G., *et al.* (2010) Representing ontogeny through ontology: a developmental biologist's guide to the gene ontology. *Mol Reprod Dev*, **77**, 314-329.
4. Mutowo-Meullenet, P., Huntley, R.P., Dimmer, E.C., *et al.* (2013) Use of Gene Ontology Annotation to understand the peroxisome proteome in humans. *Database : the journal of biological databases and curation*, **2013**, bas062.
5. Ochs, M.F., Peterson, A.J., Kossenkov, A., *et al.* (2007) Incorporation of gene ontology annotations to enhance microarray data analysis. *Methods Mol Biol*, **377**, 243-254.
6. Burge, S., Kelly, E., Lonsdale, D., *et al.* (2012) Manual GO annotation of predictive protein signatures: the InterPro approach to GO curation. *Database : the journal of biological databases and curation*, **2012**, bar068.
7. Barrell, D., Dimmer, E., Huntley, R.P., *et al.* (2009) The GOA database in 2009--an integrated Gene Ontology Annotation resource. *Nucleic Acids Res*, **37**, D396-403.
8. Balakrishnan, R., Harris, M.A., Huntley, R., *et al.* (2013) A guide to best practices for Gene Ontology (GO) manual annotation. *Database (oxford)*, **2013**, bat054.
9. Li, D., Berardini, T.Z., Muller, R.J., *et al.* (2012) Building an efficient curation workflow for the Arabidopsis literature corpus. *Database (Oxford)*, **2012**.
10. Aerts, S., Haeussler, M., van Vooren, S., *et al.* (2008) Text-mining assisted regulatory annotation. *Genome Biol*, **9**, R31.
11. Arighi, C.N., Carterette, B., Cohen, K.B., *et al.* (2013) An overview of the BioCreative 2012 Workshop Track III: interactive text mining task. *Database : the journal of biological databases and curation*, **2013**, bas056.
12. Arighi, C.N., Lu, Z., Krallinger, M., *et al.* (2011) Overview of the BioCreative III Workshop. *BMC Bioinformatics*, **12 Suppl 8**, S1.
13. Li, D., Berardini, T.Z., Muller, R.J., *et al.* (2012) Building an efficient curation workflow for the Arabidopsis literature corpus. *Database : the journal of biological databases and curation*, **2012**, bas047.
14. Neveol, A., Wilbur, W.J., Lu, Z. (2012) Improving links between literature and biological data with text mining: a case study with GEO, PDB and MEDLINE. *Database : the journal of biological databases and curation*, **2012**, bas026.
15. Van Auken, K., Jaffery, J., Chan, J., *et al.* (2009) Semi-automated curation of protein subcellular localization: a text mining-based approach to Gene Ontology (GO) Cellular Component curation. *BMC Bioinformatics*, **10**, 228.
16. Wu, C.H., Arighi, C.N., Cohen, K.B., *et al.* (2012) BioCreative-2012 virtual issue. *Database : the journal of biological databases and curation*, **2012**, bas049.
17. Wei, C.-H., Harris, B.R., Li, D., *et al.* (2012) Accelerating literature curation with text-mining tools: a case study of using PubTator to curate genes in PubMed abstracts. *Database(oxford)*, bas041.

18. Wei, C.-H., Kao, H.-Y., Lu, Z. (2012) PubTator: A PubMed-like interactive curation system for document triage and literature curation. *Proceedings of the BioCreative 2012 workshop*, Washington, D.C., pp. 20-24.
19. Wei, C.-H., Kao, H.-Y., Lu, Z. (2013) PubTator: a Web-based text mining tool for assisting Biocuration. *Nucleic Acids Res*, **41**, W518-W522.
20. Neveol, A., Wilbur, W.J., Lu, Z. (2011) Extraction of data deposition statements from the literature: a method for automatically tracking research results. *Bioinformatics*, **27**, 3306-3312.
21. Raychaudhuri, S., Chang, J.T., Sutphin, P.D., *et al.* (2002 ) Associating genes with gene ontology codes using a maximum entropy analysis of biomedical literature. *Genome Research*, **12**, 203–214.
22. Daraselia, N., Yuryev, A., Egorov, S., *et al.* (2007) Automatic extraction of gene ontology annotation and its correlation with clusters in protein networks. *BMC Bioinformatics*, **8**, 243.
23. Auken, K.V., Jaffery, J., Chan, J., *et al.* (2009) Semi-automated curation of protein subcellular localization: a text mining-based approach to Gene Ontology (GO) Cellular Component curation. *BMC Bioinformatics*, **10**, 228.
24. Costanzo, M.C., Park, J., Balakrishnan, R., *et al.* (2011) Using computational predictions to improve literature-based Gene Ontology annotations: a feasibility study. *Database (Oxford)*, **2011**, bar004.
25. Park, J., Costanzo, M.C., Balakrishnan, R., *et al.* (2012) CvManGO, a method for leveraging computational predictions to improve literature-based Gene Ontology annotations. *Database (Oxford)* **2012**, bas001.
26. Rak, R., Rowley, A., Black, W., *et al.* (2012) Argo: an integrative, interactive, text mining-based workbench supporting curation. *Database (Oxford)*, **2012**, bas010.
27. Gobeill, J., Pasche, E., Vishnyakova, D., *et al.* (2013) Managing the data deluge: data-driven GO category assignment improves while complexity of functional annotation increases. *Database (oxford)*, **2013**, bat041.
28. Koike, A., Niwa, Y., Takagi, T. (2004) Automatic extraction of gene/protein biological functions from biomedical text. *Bioinformatics*, **2005**, 7.
29. Cakmak, A., Ozsoyoglu, G. (2008) Discovering gene annotations in biomedical text databases. *BMC Bioinformatics*, **9**, 143.
30. Blaschke, C., Leon, E.A., Krallinger, M., *et al.* (2005) Evaluation of BioCreAtIvE assessment of task 2. *BMC Bioinformatics*, **6**, S16.
31. Lu, Z., Hirschman, L. (2012 ) Biocuration workflows and text mining: overview of the BioCreative 2012 Workshop Track II. *Database (Oxford)*, **2012**, bas043.
32. Camon, E.B., Barrell, D.G., Dimmer, E.C., *et al.* (2005) An evaluation of GO annotation retrieval for BioCreAtIvE and GOA. *BMC Bioinformatics*, **6 Suppl 1**, S17.
33. Bada, M., Eckert, M., Evans, D., *et al.* (2012) Concept annotation in the CRAFT corpus. *BMC Bioinformatics*, **13**, 161.
34. Kim, J.D., Ohta, T., Tateisi, Y., *et al.* (2003) GENIA corpus--semantically annotated corpus for bio-textmining. *Bioinformatics*, **19 Suppl 1**, i180-182.
35. Dogan, R.I., Lu, Z. (2012) An improved corpus of disease mentions in PubMed citations. *Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*. Association for Computational Linguistics, Montreal, Canada, pp. 91-99.

36. Smith, L., Tanabe, L.K., Ando, R.J., *et al.* (2008) Overview of BioCreative II gene mention recognition. *Genome Biol*, **9 Suppl 2**, S2.
37. Lu, Z., Bada, M., Ogren, P.V., *et al.* (2006) Improving biomedical corpus annotation guidelines. *Proceedings of the Joint BioLINK and 9th Bio-Ontologies Meeting*, Fortaleza, Brazil, pp. 89-92.
38. Comeau, D.C., Islamaj Dogan, R., Ciccarese, P., *et al.* (2013) BioC: a minimalist approach to interoperability for biomedical text processing. *Database : the journal of biological databases and curation*, **2013**, bat064.

# BiTeM/SIBtex group proceedings for BioCreative IV, Track 4: Gene Ontology curation

Gobeill Julien<sup>1,2,\*</sup>, Pasche Emilie<sup>3</sup>, Vishnyakova Dina<sup>3</sup> and Ruch Patrick<sup>1</sup>

<sup>1</sup> University of Applied Sciences - HEG, Library and Information Sciences Geneva, Switzerland

<sup>2</sup> SIBtex, SIB Swiss Institute of Bioinformatics, Geneva, Switzerland.

<sup>3</sup> University and Hospitals of Geneva, Division of Medical Information Sciences  
Geneva, Switzerland

\* Corresponding author: Tel: 0041 22 388 17 86, E-mail: julien.gobeill@hesge.ch

## Abstract

For the BioCreative IV Track 4, we exploited the power of our machine learning Gene Ontology classifier, GOCat. GOCat computes similarities between an input text and already curated instances in order to infer GO terms. GO Annotations (GOA) and MEDLINE are used for populating the knowledge base (almost 100000 curated abstracts). For the subtask A, we designed a state-of-the-art statistical approach, using a naïve Bayes classifier and the official training set. We also investigated exploiting GeneRIFs for an alternative forty times bigger training set, but the results were disappointing, probably because of the lack of correct negative instances. For the subtask B, we applied GOCat to the first subtask output and reached promising results, up to 0.65 for Recall at 20 with hierarchical metrics. Thanks to BioCreative IV, we were able to design a complete workflow for curation. Given a gene name and a full text, this system is able to deliver highly relevant GO terms along with a set of evidence sentences; observed performances are sufficient for being used in a real semi-automatic curation workflow.

## Introduction

The problem of data deluge in proteomics is well known: the available curated data lag behind current biological knowledge contained in the literature (1–3), and professional curators need assistance from text mining in order to keep up with the literature (4–6). One particularly time-consuming and labor-intensive task is gene function curation of a full text with Gene Ontology (GO) terms. Such curation from literature is a highly complex task, because it needs expertise in genomics but also in the ontology itself. For that matter, this task was studied since the first BioCreative challenge in 2005 (7) and is still considered as both unachieved, and long-awaited by the community (8).

Our group participated in the first BioCreative. At this time, we extracted GO terms from full texts with EAGL, a locally developed Dictionary-Based classifier (9). Dictionary-Based

approaches tend to exploit lexical similarities between the information about GO terms (descriptions and synonyms) and the input text. Such approaches are limited by the complex nature of the GO terms; identifying GO terms in text is highly challenging, as they often do not appear literally or approximately in text. Another smaller part of systems evaluated in BioCreative I relied on machine learning approaches. Such algorithms empirically learn behaviours from a knowledge base that contains training instances, i.e. instances of already curated publications. At that time, machine learning approaches produced lower results; the lack of a standard training set was notably pointed out.

We recently report on GOCat (10, 11), our new machine learning GO classifier. GOCat exploits similarities between an input text and already curated instances contained in a knowledge base to infer a functional profile. GO Annotations (GOA) and MEDLINE make now possible to exploit a growing amount of almost 100000 curated abstracts for populating this knowledge base. Evaluated on the first BioCreative benchmark, GOCat achieved performances close to human curators, with 0.65 for Recall at 20, against 0.26 for our dictionary-based system. Moreover, we showed in (11) that the quality of the GO terms predicted by GOCat continues to improve across the time, thanks to the growing number of high-quality GO terms assignments available in GOA: since 2006, GOCat performances have improved by 50%.

The BioCreative IV Track 4 was the occasion to exploit the GOCat power in a reference challenge. The subtask A aimed at evaluating system for filtering relevant sentences for GO curation, given a gene name and a full text. For this subtask, we designed a robust state-of-the-art approach, using a naïve Bayes classifier and the official training set (1346 positive sentences). We also investigated exploiting GeneRIFs for an alternative training set (76000 positive sentences). Then, the goal of the subtask B was to use these relevant sentences for assigning GO terms to the given gene. For this subtask, we submitted results computed with GOCat with different numbers of proposed GO terms.

## **Material and Methods**

### **Subtask A**

The goal of the subtask A was to determine, given a training set of curated sentences, whether new sentences are relevant for curation or not, and if possible to support the decision with a confidence score. Some state-of-the-art methods suitable for such supervised binary classification task include naïve Bayes classifiers and Support Vector Machines (SVM) (12,13). For implementation reasons, we chose a naïve Bayes classifier first, and finally did not investigate SVM due to a lack of time.

As we mentioned above with the GOCat description, we are used to work with statistical GO classification at the abstract/paragraph level, but we rarely apply our system at the sentence level.

Thus, for this subtask A, we further analysed the data in order to design a training set, and finally made some strong assumptions about them. First of all, we studied the length of evidence texts: as mentioned in the guidelines (14), the evidence texts for GO annotations may be derived from a single sentence, or multiple continuous, or discontinuous, sentences. In the training data, 66% of evidence texts contained only one sentence, 20% contained two sentences, 14% three and more. Hence, our first assumption was to consider only sentences: for example, a block of three positive sentences was considered as three independent positive sentences. Then, we compared, given a full text and a gene name, the set of the positive sentences, and the set of sentences where we were able to identify the gene name. For retrieving a given gene name in sentences, we relied on mapping patterns. With a simple case-insensitive mapping, we found the given gene name in 65% of the positive sentences. Then, we searched hyphens in gene names and generated a couple of variants (e.g. for “rft-1” we also tried to map “rft1”). With this rule, we reached 80%. We then investigated how to exploit the gene id in order to find supplementary synonyms and variants in reference databases, but we quickly concluded that this strategy would have brought too much noise. A further look to the data revealed that for most sentences in the 20% missed, the gene name was not explicit but often mentioned via pronouns, or such grammatical expressions that require a syntactic analysis and that is beyond statistical approaches. Hence, we accepted this limit, and our second assumption was to only consider sentences that contained the gene name. So, 80% of positive sentences contain the gene name. On the other hand, 20% of sentences that contain a given gene name are positive, 80% are negative (i.e. not positive). This was our third assumption: the training data should contain this 4:1 ratio, four negative sentences for one positive sentence. Finally, for the design of training data, we replaced all the gene names we identified by the word “genemention”.

We thus were able to design training sets for our naïve Bayes classifier. For the `gotaska_bitemteam_run1`, we built the training set from the official training set that contained 100 curated articles. With our assumptions, we finally obtained a set of 9251 sentences containing gene names: 1346 positives and 7905 negatives. The ratio is slightly different (85% of negatives), possibly because positive sentences can apply for several enumerated genes. For the `gotaska_bitemteam_run2`, we added the development set (50 curated articles) to the previous training set, and thus obtained 683 supplementary positive sentences and 3912 supplementary negative sentences.

Finally, we investigated a second way for designing our training set, based on GeneRIFs. GeneRIFs are concise phrases identified in journal papers and describing a protein function, recorded in the reference databases by a curator. GeneRIFs are not GO annotations, but potentially provide positive sentences for our task. We first downloaded all available GeneRIFs (<http://www.ncbi.nlm.nih.gov/gene/about-generif>). In July 2013, there were approximately 826000 entries in the database. Each entry is provided with the gene ID, the GeneRIF text, and the PMID that was used. As GeneRIFs are taken in full texts, we only considered papers whose

full text was available in PubMed Central (<http://www.ncbi.nlm.nih.gov/pmc/>). We were able to locate 76000 GeneRIFs in 48000 full texts. Thus, these 76000 GeneRIFs were considered as positive sentences. For negative sentences, we first retrieved all sentences containing the given gene names, and considerer that all non-positive sentences were negative, which is a strong assumption. We finally sampled this negative set in order to keep the 4:1 ratio between positive and negative instances. As for the first training sets, we replaced all identified gene names by “genemention”. This GeneRIFs training set was used for the gotaska\_bitemteam\_run3.

Hence, these three training sets were used to train our naïve Bayes classifier. For each sentence, each word was considered as a feature. We also add several meta-features, such as the type of section (paragraph, title, caption...), the relative position of the sentence in the full-text (an integer between 1 and 20), the percentage of common words with the abstract, and the sentence length. Once the classifier was trained, we parsed the test set. For each article and each gene, we extracted the sentences containing the gene name. Then, each sentence was sent to the classifier and obtained a class (positive or negative) and a confidence score. As only 20% of sentences containing a given gene name were positive in the training set, we chose to return only the first 20% best ranked sentences.

### **Subtask B**

The goal of the subtask B was to predict GO terms for a given gene in a given article. For this purpose, we used our GO classifier GOCat. GOCat relies on a  $k$ -Nearest Neighbors ( $k$ -NN), a remarkably simple algorithm which assigns to a new text the categories that are the most prevalent among the  $k$  most similar instances contained in the knowledge base. The GOCat knowledge base contains the nearly 100000 MEDLINE abstracts that were used for manual GO curation in the GOA database. GOCat is comprehensively described in (11).

Obviously, we discarded all the test set PMIDs from the knowledge base. Then, we started from the gotaska\_bitemteam\_run1. For each article and each gene name, we built a paragraph with the submitted sentences, then we sent the paragraph to GOCat. GOCat was used with  $k=100$ . As the  $k$ -NN usually outputs all possible GO terms along with a confidence score, we only kept the five most confident GO terms for gotaskb\_bitemteam\_run1, the ten most confident for gotaskb\_bitemteam\_run2, and the twenty most confident for gotaskb\_bitemteam\_run3.

## **Results and Discussion**

### **Subtask A**

Table 1 presents our results for the subtask A, computed with the official evaluation script, with two values used for the parameter (0 for partial match and 1 for exact match).

Run	Parameter	Precision	Recall	F1	Training set for Naive Bayes
gotaska_bitemteam_run1	0	0.344	0.213	0.263	Official training set
	1	0.206	0.128	0.158	
gotaska_bitemteam_run2	0	0.354	0.22	0.271	Official training and development set
	1	0.217	0.134	0.166	
gotaska_bitemteam_run3	0	0.204	0.127	0.156	GeneRIFs training set
	1	0.107	0.066	0.082	

**Table 1.** Official results of BiTeM SIBtex for subtask A.

The best results were obtained by the first two runs, computed with the official training and development set. The contribution of the development set in regards to performances is manifest but light: +3% for F1. These two runs were computed with a state-of-the-art statistical approach, relying on simple and strong – thus robust – assumptions, and the use of a simple binary classifier. At this stage, we don't know the others participants' results, so it is difficult to situate our performance. But we can compare the first two runs and the third one, which used GeneRIFs as training data. This third run was significantly weaker (appr. -50% for F1) while the used training set was forty times bigger. There is obviously a quality problem in the GeneRIFs training set. Its positive instances are built on the assumption that GeneRIFs are relevant sentences for GO annotation; this assumption seems *a priori* true, but maybe curators would make some distinctions between these two roles. But the weaker point seems to be the construction of the negative set. For the GeneRIFs training set, we considered that all sentences that mentioned the gene and were not positive were negative. Yet, GeneRIFs do not aim to produce an exhaustive set of evidence sentences in a paper, but only keep one sentence as evidence, while the annotation was exhaustive in the official BioCreative training set. Thus, there were 13 positive sentences per article in the BioCreative training set, against 1.6 in our GeneRIFs training set. The probability of false negatives sentences in the GeneRIFs training set is high and could mainly explain this counter-performance.

### Subtask B

Table 2 presents our results for the subtask B, computed with the official evaluation script, with two values used for last parameter (0 for standard metrics and 1 for hierarchical metrics).

Once again, at this stage we do not know the other participants' results, but we can compare the GOCat performances with the performances we observed in previous studies. In (11), GOCat was evaluated on its ability to retrieve GO terms that was associated to a given PMID, without taking account of the gene. For Recall at 20 (R20), GOCat achieved performances ranging from 0.56 for new published articles to 0.65 for BioCreative I test set. These performances were obtained by using the abstract for the input text. In this subtask B, the observed R20 is 0.306. But this performance was obtained by taking account of the gene, as the input was a set of sentences

dealing with a given gene, and the output was GO terms relevant for this gene. Anyway, these performances are beyond the maximum performances observed in (11) with Dictionary-Based approaches, which exploit similarities between the input text and GO terms themselves. Thanks to its knowledge base designed from real curated articles, GOCat is able to propose GO terms that do not appear literally or even approximately in text.

Run	Last parameter	Precision	Recall	F1	# GO terms returned
gotaskb_bitemteam_run1	0	0.117	0.157	0.134	5
	1	0.323	0.356	0.339	
gotaskb_bitemteam_run2	0	0.092	0.245	0.134	10
	1	0.248	0.513	0.334	
gotaskb_bitemteam_run3	0	0.057	0.306	0.096	20
	1	0.179	0.647	0.280	

**Table 2.** Official results of BiTeM SIBtex for subtask B.

Regarding hierarchical metrics, it is quite surprising to observe such a difference (R20 0.647, +111%), while GOCat aims at returning the GO terms that were most used by curators in GOA. Yet, this performance is remarkable, and is promising in a workflow where the curators would give the gene name and the PMID, then screen and check the proposed GO terms. In a fully automatic workflow, the best setting would be to return five GO terms. In this case, the observed F1 (0.134) still is far from human standards for strict curation, but the hierarchical F1 (0.339) seems sufficient for producing added value data. In this perspective, GOCat was used to profile PubChem bioassays (15), or within the COMBREX project to normalize functions described in free text format (16).

## Conclusion

The main limit of GOCat, both observed by reviewers and mentioned in our papers, was the difficulty to integrate it in a curation workflow: it is stated that GOCat proposes more accurate GO terms, but these terms are inferred from the whole abstract, then the curator still has to locate the function in the publication and to link the correct GO term with a gene product. Thanks to BioCreative IV, we were able to design a complete workflow for curation and to evaluate it. Given a gene name and a full text, this system is able to deliver relevant GO terms along with a set of evidence sentences; observed performances are sufficient for being used in a real semi-automatic curation workflow.

## Funding

This work was supported by the Swiss National Fund for Scientific Research (BiND project 3252B0-105755) and the FP7 program (Khresmoi project FP7 – 257528).

## References

1. Blake, J.A. and Bult, C.J. (2006) Beyond the data deluge: data integration and bio-ontologies. *J. Biomed. Inform.*, 39, 314–320.
2. Howe, D., Costanzo, M., Fey, P. et al. (2008) Big data: the future of biocuration. *Nature*, 455, 47–50.
3. Baumgartner, W., Bretonnel Cohen, K., Fox, L., Acquaah-Mensah, G. and Hunter L (2007) Manual curation is not sufficient for annotation of genomic databases. *Bioinformatics* 2007, 23:i41-i48.
4. Bodenreider, O. (2008) Ontologies and data integration in biomedicine: success stories and challenging issues. *Data Integr. Life Sci.*, 5109, 1–4. doi:10.1007/978-3-540-69828-9\_1.
5. Spasic, I., Ananiadou, S., McNaught, J. et al. (2005) Text mining and ontologies in biomedicine: making sense of raw text. *Brief. Bioinformatics*, 6, 239–251.
6. Hirschman, L., Gully, A., Krallinger, M. et al. (2008) Text mining for the biocuration workflow. *Database*, 2012, bas020.
7. Blaschke, C., Leon, E.A., Krallinger, M., Valencia, A. (2005) Evaluation of BioCreative assessment of task 2. *BMC Bioinformatics*, 6, S16.
8. Lu, Z., Hirschman, L. (2012) Biocuration workflows and text mining: overview of the BioCreative 2012 Workshop Track II. *Database (Oxford)*, 2012, bas043.
9. Ruch, P. (2006) Automatic assignment of biomedical categories: toward a generic approach. *Bioinformatics*, 22, 658–664.
10. Gobeill, J., Pasche, E., Teodoro, D. et al. (2012) Answering Gene Ontology terms to proteomics questions by supervised macro reading in Medline. In: *Proceedings of NETTAB Conference, EMBnet.journal, North America 18, Nov. 2012.*
11. Gobeill, J., Pasche, E., Vishnyakova, D. and Ruch, P. (2013) Managing the data deluge: data-driven GO category assignment improves while complexity of functional annotation increases. *Database (Oxford)*. 2013 Jul 9;2013:bat041. doi: 10.1093/database/bat041.
12. Huang, J., Lu, J. and Ling, C. (2003) Comparing Naive Bayes, Decision Trees, and SVM with AUC and Accuracy. In *Proceedings of the Third IEEE International Conference on Data Mining (ICDM '03)*. IEEE Computer Society, Washington, DC, USA.
13. Colas, F. and Brazdil, P. (2006) Comparison of SVM and Some Older Classification Algorithms in Text Classification Tasks *Artificial Intelligence in Theory and Practice*, 169-178.
14. Van Auken, K., Schaeffer, M., McQuilton, P. et al. (2013) Corpus Construction for the BioCreative IV GO Task. *BioCreative IV Proceedings*.
15. Guha, R., Gobeill, J. and Ruch, P. (2009) GOAssay: from Gene Ontology to Assays Identifiers – Towards Automatic Functional Annotation of PubChem BioAssays, Available from Nature Precedings <http://dx.doi.org/10.1038/npre.2009.3176.1>.
16. Anton, B., Chang, Y., Brown, P. et al. (2013) The COMBREX Project: Design, Methodology, and Initial Results. *PLoS Biol* 11(8): e1001638. doi:10.1371/journal.pbio.1001638.

# Integrating Information Retrieval with Distant Supervision for Gene Ontology Annotation

Dongqing Zhu<sup>1,2</sup>, Dingcheng Li<sup>1</sup>, Ben Carterette<sup>2</sup>, Hongfang Liu<sup>1</sup>

Mayo Clinic<sup>1</sup>, Rochester, MN  
University of Delaware<sup>2</sup>, Newark, DE

## Abstract

This paper describes our participation in the Gene Ontology Curation task (GO task) of BioCreative IV. In particular, we participated in both subtasks: A) identification of GO evidence sentences for relevant genes in full-text articles; and B) prediction of GO terms for relevant genes in full-text articles.

For subtask A, we take the approach of learning from positive and unlabeled data (LPU) to mitigate of the problem of limited training data. In particular, we first build multiple features and train a logistic regression model to detect GO evidence sentences. Then, we leverage dictionary look-up and gene-ontology mapping to annotate genes for the predicted positive instances. Our best performing system achieves an F1 score of 0.27 when the overlapping ratio is set to 1.0.

For subtask B, we design two different types of systems: 1) search-based systems, which predict GO terms based on existing annotations for GO evidences that are of different textual granularities (i.e., full-text articles, abstracts, and sentences) and are obtained by using state-of-the-art information retrieval techniques (i.e., a novel application of the idea of *distant supervision*); and 2) similarity-based systems, which assigns GO terms based on the distance between words in sentences and GO terms/synonyms. Our search-based system significantly outperforms the similarity-based system. Our best system achieves 0.075 on flat F1 and 0.301 on hierarchical F1.

**Keywords:** Gene ontology, annotation, information retrieval, classification

## Introduction

The gene ontology (GO) provides a set of concepts for annotating functional descriptions of genes and proteins in biomedical literature. The resulting annotated databases are useful for large scale analysis of gene products. However, performing GO annotation (GOA) requires expertise from well-trained human curators. Due to the fast growing of biomedical data, GOA becomes extremely labor-intensive and costly (1). Thus, biomedical texting mining tools that can assist GOA and reduce human effort are highly desired (1-3).

To promote research and tool development for assisting GO curation from biomedical literature, the Critical Assessment of Information Extraction in Biology (BioCreative) IV organized a Gene Ontology Curation Task (GO task) in 2013 (4). There are two subtasks: A) identification of GO evidence sentences for relevant genes in full-text articles; B) prediction of GO terms for relevant genes in full-text articles. The training set of GO task contains 100 full-text journal articles while the development and test sets each have 50 articles. Task organizers also provide ground-truth annotations for the training and development sets to all participants (5).

In this paper, we describe our systems which participate in the GO task. For subtask A, we take the approach of learning from positive and unlabeled data (LPU) to mitigate of the problem of limited training data. In particular, we first build multiple features and train a logistic regression model to detect GO evidence sentences. Then, we leverage dictionary look-up and gene-ontology mapping to annotate genes for the predicted positive instances. For subtask B, we design two different types of systems: 1) search-based systems, which predict GO terms based on existing annotations for GO evidences that are of different textual granularities (i.e., full-text articles, abstracts, and sentences) and are obtained by using state-of-the-art information retrieval techniques; and 2) similarity-based systems, which assigns GO terms based on the distance between words in sentences and GO terms/synonyms. Our search-based system significantly outperforms the similarity-based system. Our best system achieves 0.075 on flat F1 and 0.301 on hierarchical F1.

In the rest of paper, we will first describe our systems of subtask A and B in more detail. Then, we will present and discuss the official evaluation results. Finally, we draw conclusion and point possible directions for future work.

### **System Description: Subtask A**

In this subtask, given a full-text article we need to identify GO evidence sentences and annotate genes related to these sentences. One major challenge of this subtask is that the size of training set is very small. To address this problem we take the approach of learning from positive and unlabeled data (LPU) (6-8). Elkan and Noto showed that, if positive instances are obtained randomly in the application domain, an LPU classifier trained on just positive instances can perform as well as the one trained with both positive and negative instances (9). Therefore, we hypothesize that the available positive samples in the training data are random samples from biomedical domain, and we further expand the size of negative samples based on external resources. By doing so, we can obtain enough data to train a good classifier.

### **Identification of GO Evidence Sentences**

Many LPU systems adopt a two-step strategy. The first step is to obtain a reliable negative data set (RN) and the second step is to refine or augment RN using various learning methods, such as clustering and boosting. For example, Li and Liu (8) proposed a method for training a classifier

based on positives and unlabeled documents with clustering methods. In our system, we use a similar approach.

### ***Data preprocessing***

We extract positive and negative instances (i.e., sentences) from both training and developing sets to train our model. The training set produces 1318 positive and 26868 negative instances while the development set gives 558 positive sentences and 14580 negative sentences.

We use GeneRIF (10) data as an unlabeled data pool which contains excerpts from literature about the functional annotation of genes described in Entrez Gene. In particular, each record contains a taxonomy ID, a Gene ID, a PMID, and a GeneRIF text excerpt extracted from literature. We obtain about 20,000 excerpts from human GeneRIF records.

### ***Feature extractions***

Bag-of-words (BOW) features: we run MedTagger (11) on each sentence to generate a vector of stemmed words. In particular, we use 1 or 0 to indicate the presence or absence of feature words. Bigram features: bigrams in each sentence are potentially useful because we observe that many genes names are bigrams.

Section features: they indicate where the sentences are from, i.e., title, abstract, introduction, methods, discussion, and etc.

Topic features: these features are generated by Latent Dirichlet Allocations (LDA) (12), which can effectively group similar instances together based on their topics (13-16).

Presence of genes: we again use binary number 0/1 to indicate the presence of a specific gene.

### ***Model Training***

We apply logistic regression (LR) to the features for predicting labels for each instance. In particular, we impose a constraint on model parameters in a regularized logistic regression to avoid over-fitting and to improve the prediction performance on unseen instances. LR-TRIRLS which implements ridge regression is used with five-fold cross validation.

### ***Gene Annotation***

After negative testing instances are filtered out, remaining positive instances are left for gene-ID assignment. We use four main approaches in our system.

### ***Direct matching with dictionary look-up***

Basic dictionary look-up is done for each positive sentence, i.e., after we tokenize each sentence we use a simple string matching method to check whether there are genes appearing in the sentences. If so, corresponding gene IDs found in the dictionary are assigned to that sentence.

### ***Gene reference***

In the testing set, gene IDs are provided for each article. Based on the gene ID and gene families data we retrieve a list of gene synonyms. If a synonym is found in a sentence, we assigned the corresponding gene ID to that sentence.

a. Assignments based on coreference resolution and distance measures:

We hypothesize that adjacent sentences in the same section have high similarity, especially for those having coreferring expressions. Therefore, if a coreference is found among sentences or sentences appear close to each other, gene IDs assigned to one of those sentences are assigned to others as well.

b. Assignments based on gene-sentence distributions:

We find that there are still 32 genes to which we cannot find related sentences by using the above method a. We hypothesize that not all sentences in the article would be related to some genes. Instead, probability distributions over words for those missing genes can be obtained from the training and development data and used for identifying gene-related sentences.

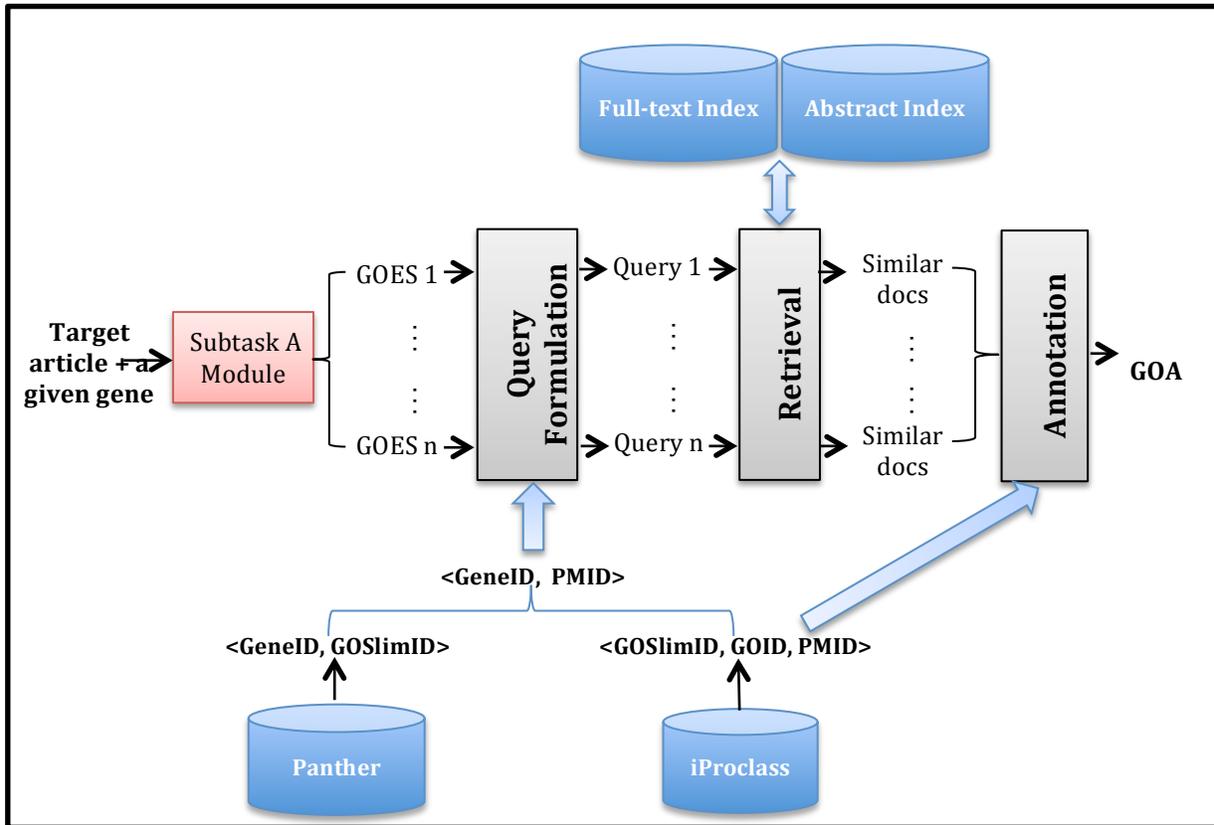
### **Submissions**

For the three systems we submit, we use thresholds of dividing positive and negative instances computed with the logistic regression model. As is known, as a binary classification model, logistic regression would yield values, which tell how much probability that one instance would be a negative as well as a positive. Then, thresholds can be selected for the whole data as a boundary of positives and negatives. It is found that the threshold of 0.1 would make the F-measure highest. Meanwhile, in the testing, we combine training and development data, including positive and negative instances with additional negatives from GeneRIF. Due to the large number of negative instances in GeneRIF, they are selected by sampling. Run 1 is based on the first sampling while run 2 is on the second sampling. The combination of the first run and the second run forms run 3.

## **System Description: Subtask B**

### **Search-based Annotation: Systems B1 & B2**

In this section, we describe two systems that generated the first two runs of Task B. The basic idea is that we want to leverage existing GO annotations (GOA) to label new articles. In particular, we search for relevant documents (sentences or abstracts or full-text articles) that have



**Figure 1.** Overview of the System

existing GOA to the target article, and then score and aggregate these existing GOA to produce the GOA for the target article.

### System B1

Figure 1 gives an overview of System B1. We highlight external resources with blue color and system modules with grey color. Next, we describe each part in detail.

### Resources

We use the following external resources:

1. Panther (17), from which we build  $\langle \text{GeneID}, \text{GOSlimID} \rangle$  pairs.
2. iProclass (18), from which we obtain  $\langle \text{GOSlimID}, \text{GOID}, \text{PMID} \rangle$  triplets.
3. A collection of PMC full-text articles that serve as the source for finding relevant documents.
4. A collection of PubMed abstracts, used as a complementary source retrieving. This is because only abstracts are publically available for some MEDLINE articles.

### ***Retrieval***

We build indexes for the abstract collection and the full-text collection respectively by using the Indri (19) search engine. In particular, we use the Porter stemmer for stemming words in the documents.

We choose the query likelihood language model as our retrieval model. This model scores documents for queries as a function of the probability that query terms would be sampled (independently) from a bag containing all the words in that document. Formally, the scoring function is a sum of the logarithms of smoothed probabilities:

$$\text{score}(D, Q) = \log P(Q|D) = \sum_{i=1}^n \log \frac{tf_{q_i,D} + \mu \frac{tf_{q_i,C}}{|C|}}{|D| + \mu}$$

where  $q_i$  is the  $i^{\text{th}}$  query term,  $|D|$  and  $|C|$  are the document and collection lengths in words respectively,  $tf_{q_i,D}$  and  $tf_{q_i,C}$  are the document and collection term frequencies of  $q_i$  respectively, and  $\mu$  is the Dirichlet smoothing parameter. The Indri retrieval engine supports this model by default.

### ***Query Formulation***

We formulate a query for each detected gene ontology evidence sentence (GOES) from the output of subtask A. In particular, we filter stop words in the sentence using a standard stop list. We leverage information in <GeneID, GOSLIM, GO> triples to reduce the GO candidate list (denoted as C), and then build a PMID candidate list by incorporating information in the <PMID, GOA> pairs. The following lists the detailed steps:

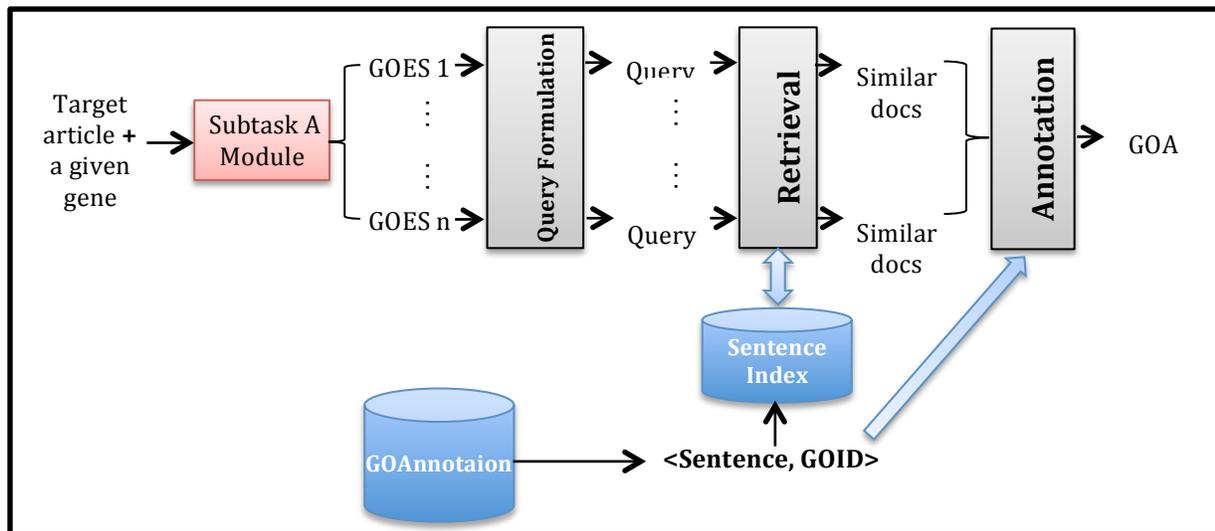
1. Given a gene G, we have a list of <G, GOES> pairs.
2. For each <G, GOES> pair, we find the corresponding <G, GOSlimID> pairs.
3. For each <G, GOSlimID> pair, we get a list of PMIDs based on <GOSlimID, GOID, PMID> triplets.
4. Combine all PMIDs for G to get a <G, L> pair, where L is the PMID candidate list (a reduced searching list) for G.

We implement query formulation in Indri.

### ***Annotation***

The output from the retrieval model for a given <GeneID, GOES> pair is a list of documents ranked by their relevance scores. Based on the <GOSlimID, GOID, PMID> triplets, we obtain GOIDs for top-ranked  $k$  documents, and then weight each GOID by their corresponding document relevance score. We further aggregate scores of each GOID and take the top-ranked  $m$  GOID for each GOES. Finally, we combine GOID across all GOES, rank them according to their occurrences, and keep GOID which occurs more than  $p$  times. For our submission, we set  $k$ ,

$m, p >$  to  $\langle 7, 10, 4 \rangle$  by training them on the 150 articles (i.e., the combination of training and development sets).



**Figure 2.** Overview of System 2

### System 2

Figure 2 gives an overview of system 2 which has similar modules to System 1. The major difference is that we use the existing GeneRIF (10) as the external resource. In particular, we extract  $\langle \text{Sentence}, \text{GOID} \rangle$  pairs from GeneRIF and build an index for this collection of sentences. Therefore, the output from the Retrieval model is a ranked list of sentences, which we further convert to a ranked list of GOID based on  $\langle \text{Sentence}, \text{GOID} \rangle$  pairs. Finally, in the Annotation module we do the following:

1. Starting from an initial list that contains top-ranked  $k$  GOID, select GOID one by one down the list until the score difference of current GOID with the topmost GOID is above threshold  $h$ .
2. Aggregate GOID frequency across all GOES associated with a particular gene, and rank GOID by frequency.
3. Take the top-ranked  $m$  GOID for each gene.

Again, based on training results we set  $\langle k, h, m \rangle$  to  $\langle 5, 0.1, 3 \rangle$ .

### Similarity-based Annotation: System B3

We use a greedy string matching algorithm which obtains all words in the sentences that are aligned to GO terms and synonyms when ignoring lexical variations. We then compute the Jaccard distance between those matched words with GO terms and synonyms. A threshold of 0.75 was used for GO term assignment.

## Results and Discussion

### Subtask A

#### 1. Feature contributions

During the development, we test feature contributions with LPU classifications. We find that all the features (unigrams, bigrams, gene existence, section, and topic features) lead to performance improvement over the baseline. In particular, section feature, as the biggest contributor, improves the F1 score by 0.01. Bigram and gene presence features each brings an improvement of 0.008. Topic features further adds 0.003 when the number of topics is set to 100.

#### 2. GeneID annotation

Gene ID annotations are the goal of subtask A. We submitted results with three systems (A1-A3), which in fact utilize the same approaches. However, system A1 and A2 use different sets of negative instances sampled from the external data while system A3 combines the samples from system A1 and A2. System A3 achieved the best results when the overlap ratio is 0.0. However, the performances of systems A1 and A3 are almost the identical (a 0.001 difference).

In addition, system A3 has the highest precision while lowest recall, which indicates that more negative training instances lead to fewer false positives. Overall, we obtain comparable results with systems in previous challenges (20).

**Table 1.** Official evaluation results for subtask A.

System	Overlap: 0.0			Overlap: 1.0		
	Precision	Recall	F1	Precision	Recall	F1
1	0.503	0.313	0.386	0.352	0.219	0.270
2	0.442	0.314	0.367	0.310	0.220	0.257
3	0.524	0.307	0.387	0.366	0.214	0.270

### Subtask B

Table 2 presents the official evaluation results of subtask B. Our search-based systems (i.e., B1 and B) outperform the similarity-based systems (i.e., B3) significantly, though the Flat-F1 scores for both types of systems are below 0.1. However, System B1 achieves 0.301 for Hierarchical-F1.

Since the results have not been distributed among all participants, we are not able to compare our systems with those from other participating groups at each stage.

The performance of system A for subtask A can have a big impact on the performance of systems B because the latter uses the output of the former. Therefore, to explore the full potential of our search-based methods for subtask B, we need to use a perfect output from subtask A. However, due to time constraint we decide to leave this interesting investigation for our future work.

**Table 2.** Official evaluation results for subtask B.

System	Flat			Hierarchical		
	Precision	Recall	F1	Precision	Recall	F1
B1	0.149	0.050	0.075	0.464	0.223	0.301
B2	0.076	0.073	0.074	0.265	0.199	0.227
B3	0.039	0.026	0.031	0.312	0.167	0.217

## Conclusion and Future Work

In this BioCreative challenge, we investigated the learning method from positive and unlabeled data for detecting sentences with potential gene mentions and then utilize dictionary look-up and gene-ontology matching to annotate sentences with Gene-IDs. In addition, we exploited information retrieval techniques for finding relevant existing GO annotations and used them for assigning GO to new articles.

The results look promising compared with previous challenges. However, there is still much room for improvements. In future work, we will explore methods, such as improving the text modeling with topic modeling and deep learning, which can detect gene mentions among texts more accurately, discover more discriminative features and develop language models for searching with more accurate similarity measures.

## Funding

The work was supported by ABI: 0845523 from United States National Science Foundation and R01LM009959 from United States National Institute of Health.

## References

1. Lu, Z. and L. Hirschman. (2012) *Biocuration workflows and text mining: overview of the BioCreative 2012 Workshop Track II*. Database: the journal of biological databases and curation.
2. Van Auken, K., et al. (2009) *Semi-automated curation of protein subcellular localization: a text mining-based approach to Gene Ontology (GO) Cellular Component curation*. BMC bioinformatics. 10(1): p. 228.
3. Blaschke, C., et al. (2005) *Evaluation of BioCreAtIvE assessment of task 2*. BMC bioinformatics. 6(Suppl 1): p. S16.

4. Yuqing Mao, K.V.A., Donghui Li, Cecilia N. Arighi, Zhiyong Lu. (2013) *The Gene Ontology Task at BioCreative IV*. in *the BioCreative IV workshop*. Bethesda, Maryland.
5. Auken, K.V., Schaeffer, M.L., McQuilton, P., et al. (2013) *Corpus Construction for the BioCreative IV GO Task*. in *the BioCreative IV Workshop*. Bethesda, Maryland, USA.
6. Chen, Y., et al. (2011) *Learning from positive and unlabeled documents for automated detection of alternative splicing sentences in medline abstracts*. in *2011 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBMW)*. IEEE.
7. Liu, H., et al. (2010) *Learning from positive and unlabeled documents for retrieval of bacterial protein-protein interaction literature*, in *Linking Literature, Information, and Knowledge for Biology*. Springer. p. 62-70.
8. Li, X. and B. Liu. (2003) *Learning to classify texts using positive and unlabeled data*. in *IJCAI*.
9. Elkan, C. and K. Noto. (2008) *Learning classifiers from only positive and unlabeled data*. in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM.
10. NIH. *GeneRIF: Gene Reference into Function*. (2013) [cited 2013 Sept. 12, 2013]; Available from: <http://www.ncbi.nlm.nih.gov/gene/about-generif>.
11. Hongfang Liu, K.W., Siddartha Jonnalagadda, Sunghwan Sohn. (2011) *Integrated cTAKES for Concept Mention Detection and Normalization*. in *Conference and Labs of the Evaluation Forum*. Valencia, Spain: Spinger.
12. Blei, D.M., A.Y. Ng, and M.I. Jordan. (2003) *Latent Dirichlet allocation*. *Journal of Machine Learning Research*, 3: p. 993-1022.
13. Li, D., et al. (2012) *Ontology-Based Temporal Relation Modeling with MapReduce Latent Dirichlet Allocations for Big EHR Data*. in *Second International Conference on Cloud and Green Computing (CGC)*, IEEE.
14. Li, D., S. Somasundaran, and A. Chakraborty. (2011) *A combination of topic models with max-margin learning for relation detection*. in *Proceedings of TextGraphs-6: Graph-based Methods for Natural Language Processing*. Association for Computational Linguistics.
15. Li, D. (2012) *Entity Relation Detection with Factorial Hidden Markov Models and Maximum Entropy Discriminant Latent Dirichlet Allocations*. UNIVERSITY OF MINNESOTA.
16. Hong, L., A.S. Doumith, and B.D. Davison. (2013) *Co-factorization machines: modeling user interests and predicting individual decisions in Twitter*. in *Proceedings of the sixth ACM international conference on Web search and data mining*. ACM.
17. *Pantherdb Classification System*. [cited 2013 Sept. 26]; Available from: <http://www.pantherdb.org/panther/ontologies.jsp>.
18. *iProclass*. Sept. 26, 2013 [cited 2013 Sept 26]; Available from: <http://pir.georgetown.edu/pirwww/dbinfo/>.
19. *Indri*. [cited 2013 Sept. 26]; Available from: <http://sourceforge.net/p/lemur/wiki/Indri/>.
20. Camon, E.B., et al., (2005) *An evaluation of GO annotation retrieval for BioCreAtIvE and GOA*. *BMC bioinformatics*. 6(Suppl 1): p. S17.

# Unsupervised Information Extraction for Finding Gene Functions

Ehsan Emadzadeh<sup>1</sup>, Azadeh Nikfarjam<sup>1</sup>, Rachel E. Ginn<sup>1</sup> and Dr. Graciela Gonzalez<sup>1</sup>

<sup>1</sup>Department of Biomedical Informatics - Arizona State University

## Abstract

Finding gene functions discussed in a literature is imperative to information extraction from biomedical documents. Automated, computational methodologies can reduce the need for manual curation significantly and improve quality of other related Information Extraction (IE) systems. We propose an open information extraction method for BioCreative IV GO shared task (Subtask b)—a workshop designed to find gene function terms (GO terms) for different genes in an article. The proposed open IE approach is based on distributional semantic similarity over the gene ontology terms. The method does not require the annotated data for training, which makes it highly generalizable. We achieve the f-measure of 0.26 for test-set in the official submission for BioCreative-GO shared task.

## Introduction

Text mining biomedical literature aims to reduce manual labor and provide more enriched information to empower research and medical treatments. Lu et al. (1) demonstrated that there is an increasing interest to use text mining techniques for curation workflows. Currently, literature curation struggles with a lack of automated annotation techniques--particularly for gene ontology annotations (1). As medical technology advances and more curation sources become available, this need magnifies. In medical informatics alone, the number of indexed articles has increased by an average of 12% each year between 1987 and 2006 (2). With an increasing number of publications detailing even more complex information, the need to have reliable and generalizable computational techniques increases rapidly.

Finding gene functions discussed in literature is crucial to genomic information extraction. Currently, tagging the gene functions in published literature is a mainly manual process. The curators find gene function evidence by reviewing each sentence in the article and mapping the results to gene ontologies. Gene Ontology (GO) (3) is a set of controlled vocabulary that defines gene product functions. BioCreative IV is a National Institutes of Health (NIH) workshop which aims to automate gene functional curation through computational methods. With a focus on gene functions, it includes two sub tasks: a) Retrieving GO evidence sentences for relevant genes, b) Predicting GO terms for relevant genes. We focus on sub task b, which finds the related gene functions (GO terms) in a set of genes discussed in an article. More details about the shared task and the corpus can be found in Auken et al. (4). This task is very similar to BioCreative I subtask

2.2 which was held in 2004 (5). Blaschke et al. (5) summarized the results for BioCreative I. For subtask 2.2 the highest precision was reported to be 34.62% (6). BioCreative IV GO subtask 2 includes an annotated corpus which enables to measure recall and f-measures. Couto et al. (7) used the IR technique to find related sentences and GO terms. Chiang et al. (6) combined sentence classification with pattern mining. Ray et al. (8) proposed a solution based on probabilistic model and Naïve Bayes classifier. Most of the participants in the previous related task focused on information content and statistical models combined with machine learning. Here, we propose an unsupervised method based on distributional semantic similarity that can be easily applied for different types of texts and ontologies.

## Material and Methods

Our method is based on distributional semantic similarity of sentences to GO terms. We use semantic vectors package (9) implementation of LSA (10) with random indexing (11) to calculate semantic similarities. GO terms' semantic vectors are created based on GO names defined in GO; one semantic vector is created for each term in the ontology. Stop-words are removed from GO name and they are generalized by Porter stemming (12).

Figure 1 shows the overall flow of our proposed method. After creating GO semantic vectors, the question is to find whether or not a sentence is related to a gene. We do this by using lexical patterns and generalizing the sentence and gene symbol (e.g. removing the numbers and non alphabetic characters). If “Sentence Gene Matcher” predicts that a sentence is related to a gene, then we calculate semantic similarity of the sentence to all GO terms using already generated semantic vectors. The “Go Finder” module finds all related GO terms to the sentence and generates the triplet of sentence, gene and GO term. Finally the output in the shared task expected format is generated by “BioC output generator”. In next section we explain about the “GO Finder” module in more details.

### “GO Finder” Module

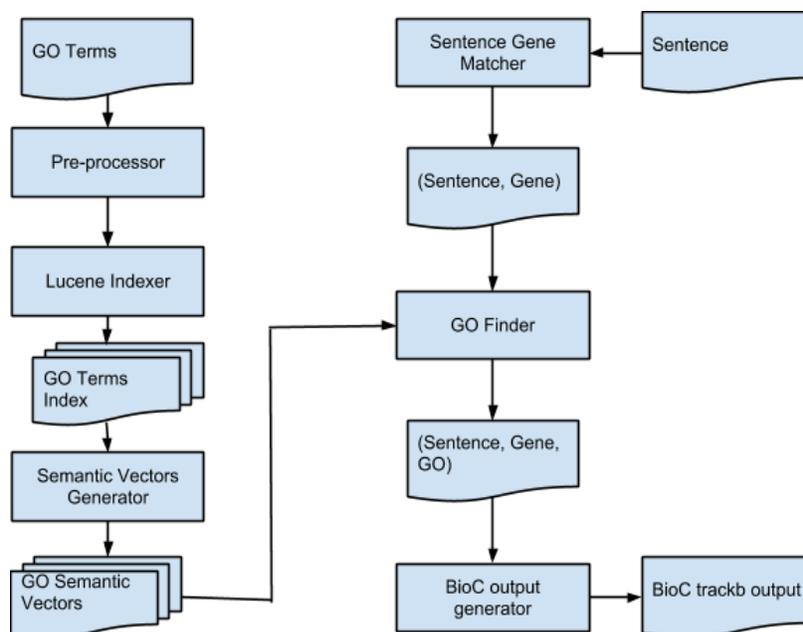
GO Finder finds related GO terms for each sentence. We define  $G$  as a set of top  $m$  GO terms with highest semantic similarity to the sentence.  $D$  is the set of top  $n$  GO terms with high similarity to the abstract of the related article. The following function returns top  $k$  similar GO terms for a given query:

$$TopSimilarGO(query, k) = \{x | x \in GOTerms \wedge |\{y \in GOTerms | Sim(x, query) < Sim(y, query)\}| < k\}$$

And  $G$  and  $D$  sets are:

$$G(sentence) = TopSimilarGO(sentence, m)$$

$$D(abstract) = TopSimilarGO(abstract, n)$$



**Figure 1** - The high level flow of the proposed system.

If a sentence is predicted to have the gene mention, the predicted GO terms for the sentence and gene are the conjunction of top similar GO terms to the sentence (set G) and top similar GO terms to the related abstract (set D):

$$\begin{aligned}
 &GeneGO(gene, sentence, abstract) \\
 &= \{G(sentence) \cap D(abstract)\} \text{ if } HasGene(sentence, gene) \text{ else } \{\}
 \end{aligned}$$

A GO term with the highest semantic similarity to the sentence in GeneGO set will be chosen as the final GO annotation for each gene in the sentence. For example if a sentence top  $m(=2)$  similar GO terms are  $\{g5, g10\}$  and the abstract top  $n(=5)$  GO terms are  $\{g4, g8, g5, g2, g9\}$ , then the final predicted GO terms for the sentence related to the gene will be  $\{g5\}$ .  $m$  and  $n$  are tuning parameters that control precision and recall.

Table 2 summarizes the number of sentences in the training set which was detected by “Sentence Gene Matcher” as relevant to a gene and also annotated to have a gene function. The table shows that “abstract”, “front” and “title2” of each document are the most important sections that might include gene function. We found that the first sentences of paragraphs have information about GO terms, but including all sentences in a paragraph will significantly reduce the precision. Therefore, we limit searching for the gene functions to the mentioned sections of the article. We choose one set of values for  $m$  and  $n$ , for “Front”, “Abstract” and “Title2” ( $m-FAT, n-FAT$ ), and choose a different set for the first sentence of the paragraphs ( $m-Paragraph, n-Paragraph$ ). Next section shows detail analysis of the impact of the tuning parameter on precision and recall.

Passage	With Gene Function	Total	Percent
front	26	67	39%
title_2	149	797	19%
abstract	225	1253	18%
paragraph	1700	20703	8%
fig_title_caption	17	412	4%
fig_caption	99	6009	2%
table_title_caption	0	47	0%
title_1, title_3, title_4	0	26	0%

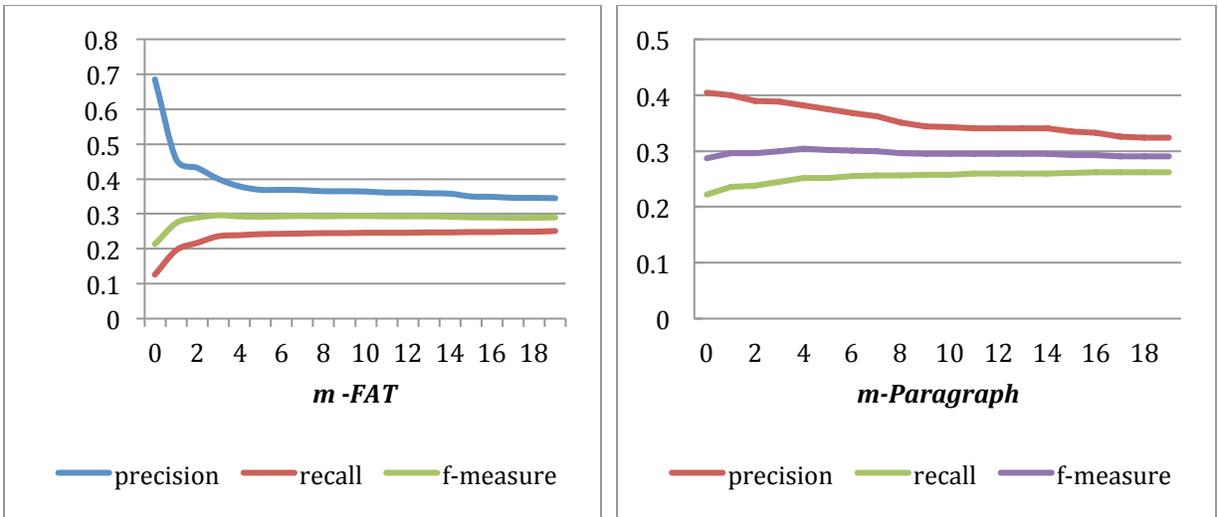
**Table 2.** Summarizes the number of sentences in the training set which was detected by “Sentence Gene Matcher” as relevant to a gene and also annotated to have a gene function.

## Results and Discussion

To achieve the highest f-measure, tuning parameters ( $m$  and  $n$ ) needs to be adjusted accordingly. We use two sets of values for  $m$  and  $n$ ; one set for the first sentence of each paragraph and another for other passage types (“abstract”, “front” and “title2”). Figure 2-a depicts precision, recall and f-measure change in respect to  $m$ -FAT changes. As  $m$ -FAT increases, precision declines and recall increases. The maximum f-measure is achieved for  $m$ -FAT=3. Therefore we assign  $m$ -FAT to 3, and try to find the best value for  $m$ -Paragraph. Figure 2-b shows the change of performance based on change of  $m$ -Paragraph. The best f-measure of 0.304 is achieved for  $m$ -FAT=3 and  $m$ -Paragraph=4.

When  $m$ -Paragraph varies, the change in f-measure is not as significant as when  $m$ -FAT varies. In addition, recall is almost constant for  $m$ -FAT >3. This shows that considering more than 4 GO terms for each sentence in FAT sections does not help us much and can only decrease the precision. On the other hand, considering only one top GO term for the first sentence of each paragraph gives the maximum boost to the recall.

We have improved parameter tuning after official submission and Figure 2 shows slightly better results than the official submission. In the first run, we tried to get the high f-measure; for run 2 and 3 we tried to get high precision and high recall respectively. Table 2 shows values of tuning parameters for each run.



**Figure 2** – a) Left diagram depicts precision, recall and f-measure change in respect to  $m-FAT$  (“Front”, “Abstract” and “Title”) changes when  $m-Paragraph=1$ . b) Right diagram shows the change of performance based on changes of  $m-Paragraph$  when  $m-FAT=3$ .

	$m-FAT$	$n-FAT$	$m-Paragraph$	$n-Paragraph$	<i>Dev set</i>	<i>Test set</i>
Run 1	3	100	1	10	0.40/0.24/ <b>0.30</b>	0.27/0.25/0.26
Run 2	3	50	0	0	<b>0.43</b> /0.20/0.27	<b>0.29</b> /0.21/ 0.24
Run 3	5	50	5	50	0.13/ <b>0.27</b> /0.18	0.25/ <b>0.29</b> / <b>0.27</b>

**Table 3.** Values of tuning parameters for 3 official runs. Run 1 targets the highest f-measure. Run 2 and 3 tries to get the highest precision and recall respectively. The numbers are precision/recall/f-measure

In this work we proposed an unsupervised approach for gene function extraction from documents. Here we only use GO terms’ names for creating semantic vectors. We tried using GO terms description but it does not help. Using more fine tuned vocabulary set for each GO term can result in more accurate vectors and probably increases the performance of this method. In addition, using term-term semantic similarity for expanding sentence terms can be evaluated. In this work we used annotations for finding the important passage types, evaluating the method and finding the best settings for the parameters. The main advantage of using unsupervised open IE technique is that it can easily be generalized and applied to similar relation extraction problems. The results from this method can be used as a baseline for supervised systems. In the future, we plan to combine this approach with supervised techniques.

## Funding

This work was partially supported by National Library of Medicine under [grant number R01LM011176]. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Library of Medicine.

## References

1. Lu,Z. and Hirschman,L. (2012) Biocuration workflows and text mining: overview of the BioCreative 2012 Workshop Track II. *Database : the journal of biological databases and curation*, **2012**, bas043.
2. Deshazo,J.P., Lavallie,D.L. and Wolf,F.M. (2009) Publication trends in the medical informatics literature: 20 years of “Medical Informatics” in MeSH. *BMC medical informatics and decision making*, **9**, 7.
3. Consortium,T.G.O. (2000) Gene Ontology: tool for the unification of biology. *nature genetics*, **25**, 25–29.
4. Auken, K.V., Schaeffer, M.L., McQuilton, P., et al. (2013) Corpus Construction for the BioCreative IV GO Task. In *Proceedings of the BioCreative IV workshop, Bethesda, USA*.
5. Blaschke,C., Leon,E.A., Krallinger,M. and Valencia,A. (2005) Evaluation of BioCreAtIvE assessment of task 2. *BMC bioinformatics*, **6 Suppl 1**, S16.
6. Chiang,J. and Yu,H. (2004) Extracting functional annotations of proteins based on hybrid text mining approaches. *Proc BioCreAtIvE Challenge Evaluation ...*
7. Couto,F.M., Silva,M.J. and Coutinho,P.M. (2005) Finding genomic ontology terms in text using evidence content. *BMC bioinformatics*, **6 Suppl 1**, S21.
8. Ray,S. and Craven,M. (2005) Learning statistical models for annotating proteins with function information using biomedical text. *BMC bioinformatics*, **6 Suppl 1**, S18.
9. Widdows,D. and Cohen,T. (2010) The Semantic Vectors Package: New Algorithms and Public Tools for Distributional Semantics. In *2010 IEEE Fourth International Conference on Semantic Computing*. IEEE, pp. 9–15.
10. Deerwester,S., Dumais,S.T., Furnas,G.W., Landauer,T.K. and Harshman,R. (1990) Indexing by latent semantic analysis. *Journal of the American society for information science*, **41**, 391–407.
11. Pentti Kanerva,J.K. Random Indexing of Text Samples for Latent Semantic Analysis.
12. Porter,M.F. (1993) An algorithm for suffix stripping. *Program: electronic library and information systems*, **14**, 130–137.

# A Robust Data-Driven Approach for BioCreative IV GO Annotation Task

Yanpeng Li<sup>1,3\*</sup>, Abhyuday Jagannat<sup>2</sup> and Hong Yu<sup>1,2</sup>

<sup>1</sup>Department of Quantitative Health Sciences, University of Massachusetts Medical School, Worcester, MA, USA

<sup>2</sup>Department of Computer Science, University of Massachusetts, Amherst, MA, USA

<sup>3</sup>Department of Computer Science, Dalian University of Technology, Dalian, China

\*Corresponding author: E-mail: liyanpeng.lyp@gmail.com

## Abstract

This paper presents our work in BioCreative IV Gene Ontology (GO) Annotation Task. For the evidence sentence extraction subtask, we built a binary classifier to identify evidence sentences using reference distance estimator (RDE), a recently proposed semi-supervised learning method that learns new features from around 10 million unlabeled sentences, achieving an F-score of 19.3% in exact match and 32.5% in relaxed match. In both development and test sets, RDE achieved much better F-score and AUC than SVM and Logistic regression. For the GO term prediction subtask, we developed an information retrieval (IR) based method to retrieve the GO term most relevant to each evidence sentence using a ranking function that combined cosine similarity and the frequency of GO terms in documents, and a filtering method based on high-level GO classes. The best performance of our submitted runs was 7.8% F-score and 22.2% hierarchy F-score. In the post-submission evaluation, we obtained a 10.6% F-score using a slightly different configuration. We found that the incorporation of frequency information and hierarchy filtering substantially improved the performance against classical language model based method. In addition, the experimental analysis showed our approaches were robust in both the two tasks.

## Subtask A – Retrieving GO evidence sentences for relevant genes

### Method

**Table 1.** Corpus statistics of the binary classification task.

	Training data	Development data
# of positive instances	965	665
# of negative instances	4255	2400

Our method includes the following steps: 1) the sentences that contain the gene names in the given list were identified by dictionary match. 2) The “gene sentences” was determined to be evidence or not by a classifier trained with annotated sentences, where each sentence with the overlap with the gold standard passages was considered as a positive instance (Table 1). 3) Each evidence sentence together with each gene ID that appears in the sentence was submitted as a positive prediction. The first and third steps are all straightforward methods for gene name normalization and linking genes to the relevant evidence sentence, since our focus is the second step, a binary classification task for distinguishing evidence and non-evidence sentence (Table 1). We found the task itself was difficult for the following reasons: 1) the sentences were not fully labeled, that is, in the annotation guideline there was no clear definition of a true negative example, so that noise would be introduced into both training and evaluation. 2) There is data sparseness problem in the classical bag-of-words representation, since a sentence contains not many words and a lot of them are low-frequency.

For the first problem, our approach of sampling the sentences that contain gene names is able to remove noisy instances to some extent, since intuitively annotators tend to check each sentence that describes the focus genes. For the second problem, we applied a semi-supervised learning method based on reference distance estimator (RDE) [1][2] to learn an enriched representation of bag-of-words features from a large number of unlabeled data. RDE is a simple linear classifier in the form of

$$f(\mathbf{x}_i, r) = \sum_j (P(r|j) - P(r))x_{ij} \quad (1)$$

where  $\mathbf{x}_i$  is the  $i$ th example represented by a Boolean vector of  $x_{ij}$ ,  $j$  is the index of feature, and  $r$  is called a reference feature. The probability of  $P(r|j) - P(r)$  can be directly estimated from unlabeled data, as long as  $r$  is not the gold standard label. In the work [1], we showed in theory that if  $r$  is discriminative to the class label and highly independent with other features, the performance of RDE tends to be close to a classifier trained with infinite labeled data. The experiment on text classification tasks showed that combining multiple RDEs from different reference features using only 5000 labeled examples performed as well as a Naïve Bayes classifier trained with 13 million labeled examples in many tasks. Therefore, the application of RDE to the GO sentence classification can be straightforward, since it is also a text classification task.

In our submitted runs, features were bag-of-words in the sentence and the paragraph, where words from sentence and paragraph were distinguished with different tags. The classifier was Algorithm 2.2 presented in the paper [1], a Logistic regression with features generated from multiple RDEs. We selected 110 reference features for semi-supervised RDE based on the labeled (training and development sets) and unlabeled data. The other parameters [1] were tuned

to optimize the performance on development data. The unlabeled data included around 10 million sentences in a subset of full text articles from the journal Science, Nature, PNAS, PLOS Genetics, Genome Research, RNA, and NAR. These full texts were downloaded under the license of the library of University of Massachusetts Medical School. Since we only sampled the “gene sentences”, for the unlabeled data we also used a gene mention recognizer [3] to get the 10 million sentences that contain gene names. Different classification thresholds were used in different submitted runs: Run 1 (threshold = 0.16), Run 2 (threshold = 0.18), and Run 3 (threshold = 0.14).

## Result

**Table 2.** Comparison of different methods on test set of Subtask A. “NER, no classifier” is the method that uses all the gene sentences as evidence sentences. SuRDE and SeRDE are the supervised and semi-supervised RDEs defined in [1]

	Precision (Exact)	Recall (Exact)	F1 (Exact)	Precision (Relaxed)	Recall (Relaxed)	F1 (Relaxed)
NER, no classifier (baseline)	9%	<b>39%</b>	14.7%	15.2%	<b>65.5%</b>	24.6%
SVM	11.1%	36.3%	17%	18.4%	60.3%	28.2%
Logistic	11.8%	33%	17.4%	19.4%	54.3%	28.6%
SuRDE	12.8%	32.6%	18.4%	20.4%	51.9%	29.3%
SeRDE (Run 1)	14.6%	28.6%	19.3%	23.9%	46.9%	31.7%
SeRDE (Run 2)	<b>15.3%</b>	25.9%	<b>19.3%</b> <b>(+31.3%)</b>	<b>25.8%</b>	43.7%	<b>32.5%</b> <b>(+32.1%)</b>
SeRDE (Run 3)	14%	31.1%	19.3%	22.6%	50.3%	31.2%

Table 2 shows the performance of different methods on the test set. The baseline is a simple rule-based method that treated all the gene sentences as evidence sentences, achieving the highest recall but lowest precision. Using different classifiers trained on the annotated corpus, the precision together with F1 improves while recall decreases. Semi-supervised RDE with the threshold 0.18 (Run 2) achieves the best performance in both exact F1 of 19.3% and relaxed F1 of 32.5%, and all the runs based on RDE achieve better F1 than SVM [4] and Logistic regression [5]. The classification thresholds of all the classifiers were tuned based on the performance on the development set, so at this level the comparison was fair. Compared with the performance on the development set in Table 3, we can see there is big improvement on the test set for supervised classifiers in both F1 measures, in particular for SVM and Logistic regression, while Semi-supervised RDE shows much more robust performance on the two different sets. Note that the task for training is binary sentence classification, while the evaluation takes into account many other factors such as gene normalization and gene-sentence linking, so the performance on test set cannot directly reflect the difference of machine learning methods. From the binary

classification task in Table 3 it is more clear to see the improvement of RDE against the other machine learning approaches, which is interestingly similar to the observation in the previous work [1].

**Table 3.** Comparison of different methods on development set of Subtask A. F1 (Exact) and F1 (Relaxed) are the official evaluation measures. The F1 (Binary) and AUC (Binary) are the performance on the binary sentence classification task defined in “Method” section and Table 1.

	F1 (Exact)	F1 (Relaxed)	F1 (Binary)	AUC (Binary)
NER, no classifier	14.6%	22.8%	-	-
SVM (baseline)	14.9%	23.4%	38.4%	62%
Logistic	15.4%	23.7%	36%	61%
SuRDE	17.9%	27.4%	45.2%	71%
SeRDE	<b>19.7% (+32.2%)</b>	<b>30.4% (+29.9%)</b>	<b>51.6% (+34.4%)</b>	<b>77.3% (+24.7%)</b>

## Subtask B – Predicting GO terms for relevant genes

### Method

Our approach was an information retrieval (IR) based framework with multiple strategies for filtering. In the method, each positive sentence in Subtask A was treated as a query, and the GO term most relevant to the sentence was returned as the candidate prediction. In the experiment, we find the frequency of GO terms has a big impact on the performance of ranking, since the occurrence of GO term in documents follows a power law distribution, that is, a small fraction of GO terms appear in a lot of documents, and most GO terms appear rarely. Therefore, if we give higher weight to the important GO terms (high-frequency terms), the F-score tend to be much better, just similar to the idea of page rank algorithm in Web search, which prefers the important pages linked by a lot of other pages. Our ranking function is:

$$GORank(sentence, GO term) = \frac{\#of\ Common\ words\ in\ sentence\ and\ GO\ term}{\sqrt{\#of\ words\ in\ sentence} \sqrt{\#of\ words\ in\ GO\ term}} \log(count(GO\ term)) \quad (2)$$

where the first part is the cosine similarity of the sentence and GO term, and  $count(GO\ term)$  is the number of documents related to the GO term in the Gene Ontology Annotation (GOA) databases (<http://www.geneontology.org/GO.downloads.annotations.shtml>). In the GORank function, both lexical similarity and frequency of GO terms are considered. In the experiment, all the words were lowercased.

In order to make use of the information in the annotated sentences to improve the performance, after the ranking, we built a classifier for 12 high-level GO classes trained on labeled sentences

to prune the result. Since there are around 40,000 GO terms in the GO database and only around 500 terms in the training data, it is difficult to build a classifier for the whole vocabulary of GO terms, but it is much easier to build a classifier for high-level GO terms, since the vocabulary becomes much smaller when moving to the root of the Ontology concept tree. According to the database ([http://archive.geneontology.org/latest-termdb/go\\_daily-termdb.rdf-xml.gz](http://archive.geneontology.org/latest-termdb/go_daily-termdb.rdf-xml.gz)), there are 3 GO terms (i.e., Cellular component, Biological process and Molecular function) in the first level, and 60 terms in the second level, of which 11 most frequent terms in training data were used to build 12 binary classifiers (one for ‘other’ class) to determine whether each sentence is related to each GO term. Here supervised RDE was used, since we did not have time to test the semi-supervised method before submission. We define a filtering threshold  $t$  as the number of  $t$  most relevant high-level GO classes to the sentence determined by the classifiers. If the highest ranked GO term by GORank is in the  $t$  classes, it will be selected as a positive result. Three submitted runs used the following different parameters:

- Run 1: use GO terms with the count over 2000 in the GOA database for ranking. The classification threshold for Sbutask A was 0. The filtering threshold was 6.
- Run 2: use GO terms with the count over 500 in the GOA database for ranking. The classification threshold for Sbutask A was 0. The filtering threshold was 8.
- Run 3: use GO terms with the count over 2000 in the GOA database for ranking. The classification threshold for Sbutask A was 0.16. The filtering threshold was 2.

**Table 4.** Performance of different methods on the test set of Subtask B. “Indri” is a language model based method [6]. “Definition” means appending the definition of GO terms to expand the text representation. “Cosine” is the similarity function in the first part of Formula (2). Frequency” is to limit GO vocabulary to the high-frequency GO terms. “Hierarchy” is the high-level GO class based filtering.

Method	Precision (Exact)	Recall (Exact)	F1 (Exact)	Precision (Hierarchy)	Recall (Hierarchy)	F1 (Hierarchy)
Indri (baseline)	1%	3%	1.5%	9.9%	33.1%	15.2%
Indri + Definition	0.8%	3%	1.3%	8.5%	34.7%	13.7%
Cosine	2.4%	7.6%	3.6%	7.2%	<b>40.6%</b>	12.2%
GORank	5.9%	<b>14.3%</b>	8.4%	13.5%	31.8%	19%
GORank + Hierarchy	<b>10.6%</b>	10.6%	<b>10.6%</b> <b>(+606.7%)</b>	21.6%	21.2%	21.4%
Cosine + Frequency	4.6%	9.8%	6.2%	15.1%	28.4%	19.7%
GORank + Frequency	5.5%	10.7%	7.3%	17.4%	27.5%	21.3%
GORank + Frequency + Hierarchy (Run 3)	9.5%	6.7%	7.8%	<b>27.8%</b>	16.1%	20.4%
GORank + Frequency + Hierarchy (Run 1)	5.2%	11.2%	7.1%	17%	32%	<b>22.2%</b> <b>(+46%)</b>
GORank + Frequency + Hierarchy (Run 2)	4.9%	14.3%	7.3	12.7%	36.8%	18.8%

## Result

Table 4 and Table 5 show the performance of various methods on the test and development data in Subtask B. As can be seen, cosine similarity performs much better than Indri [6], a classical language model based method, on exact performance but inferior on hierarchy performance. The incorporation of definition for GO term representation decreases almost all the performance. GORank outperforms both cosine similarity and Indri on most of the performance measures. Methods that incorporate the frequency of GO terms (i.e., frequency based filtering and GORank) achieve significant improvement. Run 3 achieves the best performance on exact precision and F-score on the test set. Hierarchy filtering improves the precision and F-score in both development data and test data. The simple method that uses GORank and hierarchy filtering achieves the best overall performance on test set, but not the best on development set, so this run was not submitted for the official evaluation.

**Table 5.** Performance of different methods on the development set of Subtask B

Method	F1 (Exact)	F1 (Hierarchy)
Indri (baseline)	1.3%	11.8%
GORank	5.9%	<b>17.3%(46.6%)</b>
GORank + Hierarchy	6.6%	16%
GORank + Frequency + Hierarchy (Run 3)	5.9%	12.7%
GORank + Frequency + Hierarchy (Run 1)	6.9%	16.3%
GORank + Frequency + Hierarchy (Run 2)	<b>6.9% (430.8%)</b>	16.4%

## Conclusion

We present the application of RDE based semi-supervised learning to the first subtask, and GORank with RDE based filtering for the second subtask. Our novel methods lead to big improvement on F-score and robustness against the classical text classification and information retrieval methods on the two subtasks. Nevertheless, the absolute performances of two tasks are still very low, which indicates big potential space for further study. In the future, for subtask A, we will consider using RDE with more powerful representation method, such as features from n-grams and figures concerned with the evidence experiments. For the subtask B, we think exploiting and integrating the existing annotation information in database, and the labeled sentences in BioCreative are the promising ways to make further progress.

## Funding

This work was supported by National Institutes of Health [GM095476].

## References

1. Li, Y. (2013). Reference Distance Estimator. *arXiv preprint arXiv:1308.3818*.
2. Li, Y., Hu, X., Lin, H., & Yang, Z. (2011). A framework for semisupervised feature generation and its applications in biomedical literature mining. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 8(2), 294-307.
3. Li, Y., Lin, H., & Yang, Z. (2009). Incorporating rich background knowledge for gene named entity classification and recognition. *BMC bioinformatics*, 10(1), 223.
4. Joachims, T. "Making large scale SVM learning practical." (1999).
5. Genkin, A., Lewis, D. D., & Madigan, D. (2007). Large-scale Bayesian logistic regression for text categorization. *Technometrics*, 49(3), 291-304.
6. Strohman, T., Metzler, D., Turtle, H., & Croft, W. B. (2005). Indri: A language model-based search engine for complex queries. In *Proceedings of the International Conference on Intelligent Analysis* (Vol. 2, No. 6, pp. 2-6).

# Gene Ontology Evidence Sentence Retrieval Using Combinatorial Applications of Semantic Class and Rule Patterns

Jian-Ming Chen<sup>1,\*</sup>, Yung-Chun Chang<sup>1,2</sup>, Johnny Chi-Yang Wu<sup>1</sup>, Po-Ting Lai<sup>3</sup>, Hong-Jie Dai<sup>4</sup>

<sup>1</sup>Institute of Information Science, Academia Sinica, Taipei, Taiwan, R.O.C., <sup>2</sup>Department of Information Management, National Taiwan University, Taipei, Taiwan, R.O.C., <sup>3</sup>Department of Computer Science, National Tsing-Hua University, HsinChu, Taiwan, R.O.C., <sup>4</sup>Graduate Institute of Biomedical Informatics, College of Medical Science and Technology, Taipei Medical University, Taipei, Taiwan, R.O.C.

\*Corresponding author: Tel: 886-2-27883799 Ext.1367, E-mail: jurrychen@iis.sinica.edu.tw

## Abstract

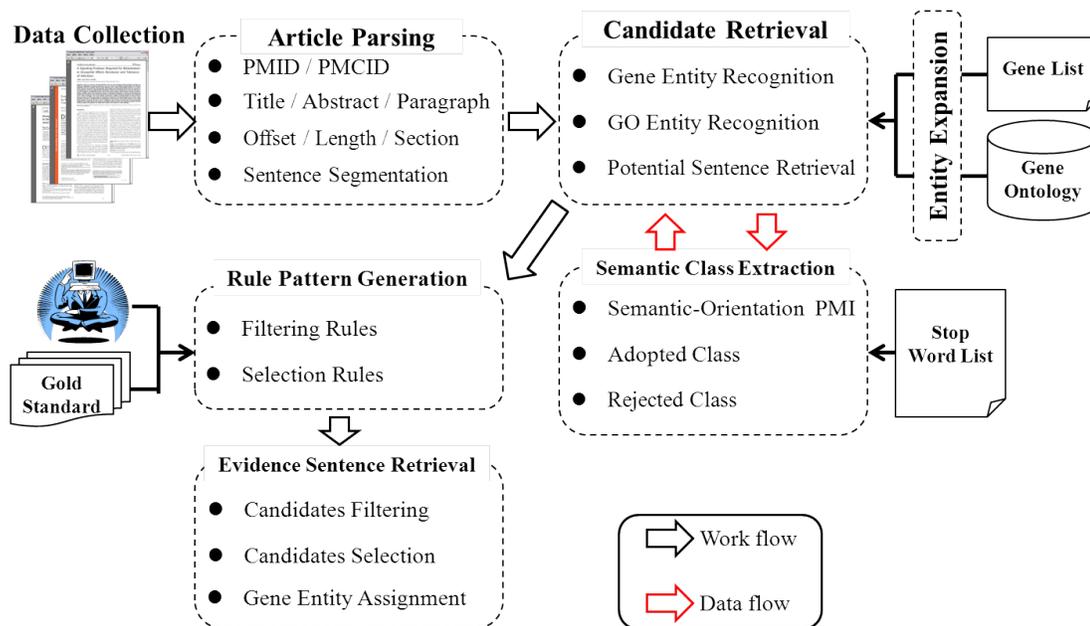
Gene Ontology (GO) provides helpful information with respect to biological process, molecular function and cellular component in annotating the relationships among gene, chemical and disease. Due to the complexity of GO knowledge, developing automated or semi-automated GO curation techniques remains to be a big challenge for database curators. In order to efficiently and precisely retrieve GO information from large amount of biomedical resources, we propose a GO evidence sentence retrieval system conducted via combinatorial applications of semantic class and rule patterns to automatically retrieve GO evidence sentences with specific gene mentions from full-length articles.

## Introduction

Gene-oriented biomedical researches constitute the basis of advanced life science researches. Although the phenomenal growth of biomedical studies augmented our apprehension of complex biological mechanisms, the sharing and exchange of these results are hindered by the discrete terminologies and depictions. Therefore, the Gene Ontology (GO) initiative attempts to provide a universal representation of gene products and their correlated attributes. To promote research and tool development for the curation of GO database, BioCreative IV hosted a GO track, with an intention of retrieving GO evidence sentences for relevant genes (SubTask A) and predicting GO terms for relevant genes (SubTask B). In this work, we introduce a combinatorial approach toward the SubTask A of BioCreative IV. In our approach, the subtask is further divided into two subtasks: 1) candidate GO sentence retrieval, which selects the candidate GO sentences from a given full text, and 2) gene entity assignment, which assigns relevant gene mentions to a GO evidence sentence.

## Material and Methods

Fig. 1 shows the workflow of our GO evidence sentence retrieval system developed for the GO SubTask A. We divide the task into the following steps: article parsing, candidate retrieval, semantic class extraction, rule pattern generation and evidence sentence retrieval.



**Figure 1.** System workflow of GO evidence sentence retrieval

### Article Parsing

Full-length articles with BioC XML format are collected and parsed before GO evidence sentence retrieval. To ensure that article information can be used efficiently and accurately during subsequent text analysis, the BioC package (1) is utilized and expanded. This package is helpful not only in parsing articles, but also in accessing article information. While datasets are parsed via BioC package, individual article metadata, including PMID, PMCID, title, abstract, paragraph, offset, length and section are received and easily accessed via function call. In order to further receive the minimized analytical unit, sentence segmentation is conducted via Apache OpenNLP (2).

### Candidate Sentence Retrieval

Sentences containing gene entities or GO terms are considered as potential evidence sentences. Before extracting the primary evidence sentences, gene entity and GO term recognition are conducted. The sources of gene entities and GO terms are derived from BioCreative IV and GO database (3), respectively. After collecting above entity terms, entity expansion is performed to enrich the contents of these terms. Finally, sentences including those terms are retrieved and deemed as the candidate GO evidence sentence set.

## Semantic Class Extraction

In this step, we apply semantic filtering and selection on the candidate evidence sentence set. Semantic classes used for filtering and selection consist of the adopted and rejected class, and both originates from semantic knowledge concealed within the training gold standard datasets. To accurately extract the semantic classes, Semantic-Orientation Point-wise Mutual Information (SO-PMI) is conducted in selecting meaningful words/terms. Words/terms within the adopted class are extracted from the training annotation file, whereas those of the rejected class are derived from the false positive sentences in the primary sentence set. The extraction of semantic words/terms and the exclusion of stop words are conducted via parameter setting in SO-PMI results. Thus, if one sentence contains a word/term from the rejected class, it will be filtered out. Likewise, if a sentence contains a word/term from the adopted class, it will be kept as a candidate GO evidence sentence. The remaining sentences after semantic filtering and selection are considered as the candidate GO evidence sentences for the next step.

## Rule Pattern Generation

**Table 1.** Rule patterns used in filtering false positive sentences

	<b>Rule Pattern</b>
[1]	[GENE] ... at ... [CELL COMPONENT]
[2]	[GENE] ... localize(s/d) to/on/at ... [CELL COMPONENT]
[3]	colocalization/localization of [GENE] to [CELL COMPONENT]
[4]	[GENE] ... on ... chromosome
[5]	[GENE] ... form(s/d) ... protein complex
[6]	[GENE] ... component of ... complex
[7]	complex ... contain ... [GENE]
[8]	distribution of [GENE] ... [CELL COMPONENT]
[9]	[GENE] ... transport(s/d) to [CELL COMPONENT]
[10]	(over)expression of ... [GENE] ... express(es/d) in [CELL COMPONENT]
[11]	[GENE] ... express(es/d) in [CELL COMPONENT]
[12]	[GENE] ... present(s/d) in [CELL COMPONENT]
[13]	... role/function of ... [GENE] ... in [GO]
[14]	[GENE]+/± ...
[15]	[GENE]-/- ...

To further maximize the performance of GO evidence sentences retrieval, rule patterns including filtering and selection rules are defined and applied in identifying sentences within the filtered sentence set. All filtering and selection rule patterns are defined and generated with the assistance of a domain expert. Complete rule patterns for filtering and selection are shown in Table 1 and 2, respectively.

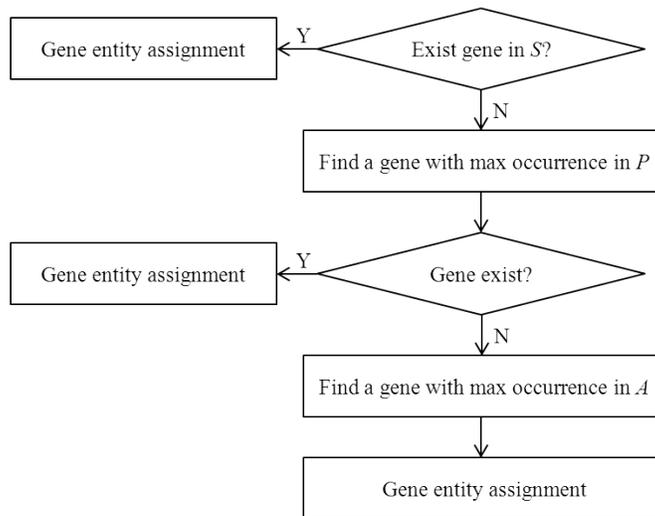
**Table 2.** Rule patterns used in selecting potential evidence sentences.

	Rule Pattern
[1]	[GENE] ... require(s/d) for ... [GO]
[2]	[GENE] ... caused/produced ... [GO]
[3]	[GENE] ... depend on/dependent/dependent on/interdependent ... [GO]
[4]	[GENE] ... regulator/regulation of ... [GO]
[5]	[GENE] ... essential for/of ... [GO]
[6]	phosphorylation/dephosphorylation ... by [GENE]
[7]	[GENE] ... lead to ... [GO]
[8]	localization of [GENE] ... to ... [GO]
[9]	[GENE] ... undergone ... [GO]
[10]	To observe/examine ... [GENE] ... [GO]
[11]	caused by ... [GENE] ... [GO]
[12]	in ... with [GENE] ... [GO]
[13]	[GO] ... in ... with [GO]
[14]	[GENE](-)depleted ... [GO]
[15]	mislocalized ... [GENE] ...
[16]	... depleted/ depletion of [GENE] ... [GO]
[17]	[GO] ... for ... [GENE]
[18]	[GENE] ... regulate(s/d) ... [GO]
[19]	[GENE] ... interferes with ... [GO]
[20]	[GENE] ... is a phosphatase/kinase

### GO Evidence Sentence Retrieval

Although sentences that remain after the previous processes contain specific GO information, they may not be properly linked to a corresponding gene mention. Therefore, the process of gene entity assignment is performed to identify the relationship between GO evidence sentences and their probable gene mentions. The complete procedure of gene entity assignment is shown in Fig. 2.

Given a sentence  $S$  and its corresponding paragraph  $P$  in article  $A$ .



**Figure 2.** The procedure of gene entity assignment.

During gene entity assignment, genes are assigned to sentence  $S$  if there exists at least one gene in  $S$ . Otherwise the procedure identify the gene with the maximum occurrence from retrieved sentences in paragraph  $P$  in which  $S$  belongs, and assign this gene to sentence  $S$ . Alternatively, identify the gene with the maximum occurrence from retrieved sentences in article  $A$ , and assign this gene to sentence  $S$ .

## Results and Conduction

Table 3 and 4 shows the performance evaluation of our Gene Ontology (GO) evidence sentence retrieval system, which is conducted via precision, recall and f-score in development datasets. In Table 3, GO evidence sentences without considering gene mentions are evaluated, while the results of Table 4 is evaluated via the evaluation tool provided by BioCreative IV.

**Table 3.** Performance evaluation without gene entity assignment

Configuration	Precision	Recall	F-Score
DEV_Candidate	0.0846	0.8652	0.1542
DEV_Filtering	0.0854	0.8366	0.1549
DEV_Selection	0.0697	0.1137	0.0865
DEV_FS	0.0691	0.1120	0.0855

**Table 4.** Performance evaluation (exact match) of development datasets

Configuration	Precision	Recall	F-Score
DEV_Candidate	0.034	0.381	0.062
DEV_Filtering	0.037	0.373	0.067
DEV_Selection	0.043	0.069	0.053
DEV_FS	0.043	0.069	0.053

In fact, Table 3 and 4 shows that GO evidence sentence retrieval using filtering rule patterns has better performance against selection rule patterns. In the future, the conduction of rule selection in rule pattern generation and co-reference resolution in gene entity assignment may be performed to maximize the overall performance.

## References

1. BioC project, <http://www.ncbi.nlm.nih.gov/CBBresearch/Dogan/BioC>.
2. The Apache Software Foundation, Apache openNLP, <http://opennlp.apache.org>.
3. The Gene Ontology consortium (2000) Gene Ontology: tool for the unification of biology. *Nature Genetics*, Vol. 25, no. 1, pp. 25-29.

# Gene Ontology Concept Recognition using Cross-Products and Statistical Methods

Luu Anh Tuan<sup>1,1</sup>, Jung-jae Kim<sup>1</sup>, See-Kiong Ng<sup>2</sup>

<sup>1</sup>School of Computer Engineering, Nanyang Technological University, Singapore

<sup>2</sup>Institute for Infocomm Research, Agency For Science Technology and Research, Singapore  
tuanluu@ntu.edu.sg, jungjae.kim@ntu.edu.sg, skng@i2r.a-star.edu.sg

## Abstract

In biomedical domain, the Gene Ontology (GO) has evolved as the standard for providing a controlled and structured vocabulary of terms describing attributes of genes and gene products, but it is still challenging to automatically locate evidence to GO terms in text. The majority of previous approaches to the automatic recognition of GO terms follow the bag-of-words model for GO term representation. In this paper, we introduce a novel approach using the GO cross-products as the GO term representation to recognize GO terms in text and show that it is complementary to a state-of-the-art method based on bag-of-words model.

## Introduction

Gene Ontology (GO) is being widely used for functional analysis in biology and bioinformatics. Since it is time-consuming to manually associate its terms with gene products, there is a need for automating the annotation (1). While there have been many efforts in developing automatic tools for GO terms recognition, this task is still a challenging problem. We present here a novel method for recognizing GO terms in text using GO cross-products and statistical methods.

One key challenge for recognizing GO terms in text is that many of them do not appear literally in biomedical texts. To address this, many previous approaches for GO term recognition represent a GO term as the bag of its component words. However, this representation can suffer low precision because it does not take into account of the syntactic and semantic relations between the component words in a GO term. To deal with this issue, we propose to utilize the cross-product extensions of Gene Ontology, which are explicit, logical representations of the compositional relations implied in GO terms (2).

The GO Cross-Product database is an effort of normalizing the GO by explicitly stating the definitions of compositional classes in a form that can be used by logical reasoners. Such a definition consists of mutually exclusive cross-products, many of which reference other OBO Foundry candidate ontologies for such entities as chemical entities, proteins, biological qualities

---

<sup>1</sup> Corresponding author

and anatomical entities. The cross-products of a GO term are combined with `intersection_of`, which list the necessary and sufficient conditions for the GO term/class. Table 1 shows an example GO term defined by cross-products, where the concept GO:0032543 is defined as the intersection of concept GO:0006412 and the concept that can ‘occur in’ concept GO:0005739.

**Table 1.** Example cross-products

[Term] id: GO:0032543 name: mitochondrial translation intersection_of: GO:0006412 ! translation intersection_of: occurs_in GO:0005739 ! mitochondrion
---

The concepts of cross-products (e.g. “translation” and “mitochondrion” in Table 1) indicate the participants and processes related to the GO term of the cross-products (e.g. “mitochondrial translation” in Table 1). These concepts of cross-products can show a different aspect of GO term recognition compared to the bag of words model, because the bag of words model does not consider phrases within GO terms.

While the cross-products are useful for GO term recognition, not all GO terms are defined with cross-products yet. To fill the gap, we combine our method based on the cross-products with a state-of-the-art statistical method based on the bag of words model (3).

## Related Work

Many approaches have been proposed for the task of GO term recognition, including dictionary-based methods (4), bag of words (BOW) methods (3,5,6), machine learning methods (7), and syntactic pattern matching methods (8,9).

Dictionary-based methods match ontology terms directly to text. They work well for short GO terms, but not for long GO terms. As explained above, BOW methods represent a GO term as the set of component words in the GO term, and because they do not consider the syntactic/semantic relations between component words, their precision is not very high. Machine learning methods learn models from corpora annotated with GO terms, but the size of such available corpora is often too small for statistically meaningful training. Syntactic pattern matching methods represent a GO term as a compositional semantic template like cross-products (2,10) and map the syntactic structures of sentences to the semantic templates through syntactic patterns. They show high precision compared to the BOW methods but are restricted to a small number of GO terms whose semantic templates are available.

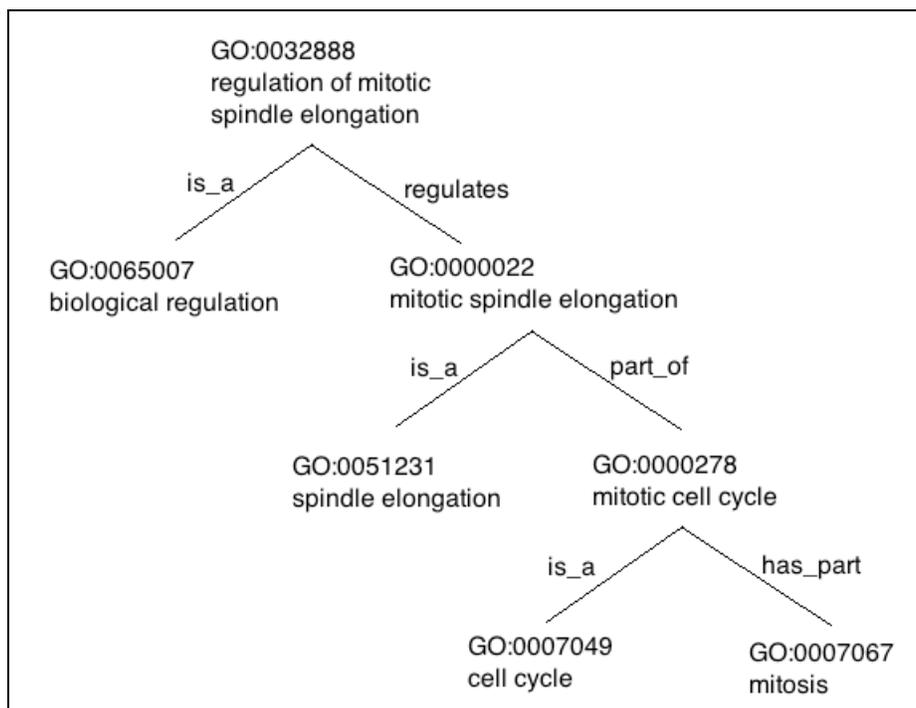
Our method is similar to the BOW approach in that it does not use any syntactic patterns nor machine learning models, but different from the approach in that it represents a GO term as the set of the concepts of its cross-products, not as the set of its component words.

## Methods

### GO term recognition based on cross-products

The Cross-Product database represents a GO term as the intersection of other ontology concepts (e.g. translation, mitochondrion) possibly with relations (e.g. occurs\_in), where we call the other ontology concepts “component concepts”. If component concepts are also defined with cross-products, we can replace them with their definitions, thus forming a tree-like definition of the original GO term. From this tree-like definition, we can collect all “primitive concepts” related to the GO term, where primitive concepts are the concepts that are not defined with cross-products. Table 2 shows an example of tree-like definition of a GO term, where the definition of the GO term “regulation of mitotic spindle elongation” has the following primitive concepts: biological regulation, spindle elongation, cell cycle, and mitosis.

**Table 2.** An example of tree-like GO term definition



Our assumption is that if all or most of the primitive concepts of a GO term appear in a small window of text (e.g. sentence), the corresponding GO term is more likely to be expressed therein. We identify the expression of a primitive concept in a text by recognizing one of the words that appear frequently in the documents that are known to express the concept (called

domain corpus), but not frequently in a representative subset of all documents (called generic corpus).

To implement this idea, we collect MEDLINE abstracts of primitive GO terms from the GOA database (1). The set of documents paired with a GO term is called the domain corpus of the GO term. Since a small number of documents are not useful for statistical analysis, if the number of documents in a domain corpus is less than  $N$ , we use the GO term as a query for PubMed and download some results of the query from the search engine to make the total number of articles  $N$  ( $N = 500$ ). A common generic corpus is used for the statistical analysis of all GO terms, which was created by collecting the 20,000 results of the query "gene" from PubMed.

We collect all words from the domain corpus of a GO term and select those with high relative frequencies. We calculate the relative frequency of a word  $\alpha$  in comparison between the domain corpus of the GO term  $\gamma$  ( $DC_\gamma$ ) and the common generic corpus (GC) as follows<sup>2</sup>:

$$f(\gamma, \alpha) = \frac{N_{DC_\gamma}^\alpha}{|DC_\gamma|} \bigg/ \frac{N_{GC}^\alpha}{|GC|}$$

where  $N_C^\alpha$  indicates the number of occurrences of the word  $\alpha$  in the corpus  $C$  and  $|C|$  indicates the total number of tokens in  $C$ .  $f(\gamma, \alpha)$  is high when the frequency of  $\alpha$  in the domain corpus is high, whereas its frequency in the generic corpus is low. In other words, the words with high  $f$  values have exclusive relationship with the GO term. Our method chooses the top- $K$  words with the highest  $f$  values as evidence words for each primitive GO concept. We chose 10 as the value of  $K$ , and show results of different values of  $K$ .

Given a document  $\delta$  and a primitive concept  $\gamma$ , if the sum of the  $f$  values of the words that are among the top- $K$  words of the concept and found in the text (designated as  $W(\delta, \gamma)$ ) is larger than a threshold  $\theta$ , we regard the concept as expressed in the document. The default value of  $\theta$  is 100. The calculation can be formulated as follows:

$$\sum_{\alpha \in W(\delta, \gamma)} f(\gamma, \alpha) \geq \theta$$

Now, we explain our method for the recognition of a GO term  $\Gamma$  whose cross-products definition has  $n$  primitive concepts. A text is considered to express  $\Gamma$  if this text expresses at least  $k$  primitive concepts among the  $n$  concepts, where the value of  $k$  is dynamically determined as follows:

---

<sup>2</sup> DC stands for domain corpus, while GC generic corpus.

$$k = n \times \left( 0.6 + \frac{1}{1 + e^{\sqrt{n}}} \right)$$

### Gaudan's method

The statistical method of (3) is based on three factors: 1) the evidence  $e$  for a GO term given by the words occurring in text, 2) the proximity  $pr$  between the words, and 3) the specificity  $I$  of the GO terms based on their information content. The three aspects are weighted and combined as follows:

$$s(z, t) = e(z, t)^4 \times I(t) \times pr(W, z)$$

where  $t$  is the term,  $z$  is the zone (e.g. paragraph, sentence) and  $W$  is the set of component words of term  $t$ .

We combine Gaudan's method with our method based on cross-products as mentioned in the Introduction, as follows: For each GO term  $\Gamma$  whose cross-products definition has  $n$  primitive concepts, if we can find evidence to  $k$  primitive concepts ( $k$  is dependent on  $n$ , as explained in Section 3.1) in the text zone  $\delta$ , we calculate the sum of the scores from the two methods as follows:

$$S(\delta, \Gamma) = s(\delta, \Gamma) + \frac{\sum_{i=1..k} \sum_{\alpha \in W(\delta, \gamma_i)} f(\gamma_i, \alpha)}{2 \times \theta \times k} (*)$$

where  $\gamma_i$  is a primitive concept of  $\Gamma$ ,  $\alpha$  is the word that is among the top- $K$  words of the primitive concept (designated as  $W(\delta, \gamma_i)$ ) and found in the text. If  $S(\delta, \Gamma)$  is greater than 0.8, we assume that  $\delta$  expresses  $\Gamma$ . If a GO term does not have a cross-products definition, we only use the score of the Gaudan's method. In short, we call the method described in (\*) XP-Gaudan method.

### Association of GO terms with gene names

We first recognize GO terms in each paragraph of a given article by using aforementioned methods and look for gene names in the paragraph, where the gene IDs found in the article are given by the organizers of the task. We use exact string match to find gene names in the paragraph. For each pair of a GO term and a gene name found in the same paragraph, we then predict an association between the two entities. This co-occurrence method for the association and the exact string match for gene name recognition are primitive, and we leave as a future work developing more sophisticated methods.

## Experimental results

In this section, we show our experimental results for the BioCreative IV GO task corpus. The corpus comprises a total of 200 full-text articles. Over 7,000 text passages were used in the annotation of 1,311 unique GO terms. Note that we apply the methods on all 36,565 GO terms listed in Gene Ontology Annotation (UniProt-*GOA*) Database.

Table 3 shows the results of the XP-Gaudan method with and without the association of GO terms with gene names. The result of the method with the association is lower than that of the method only with the GO term recognition because of the primitive methods of exact string matching and co-occurrence-based association. In particular, the exact string match shows incorrect guessing of gene names and missing gene names, which are indirectly mentioned in the text (e.g. anaphoric expressions).

**Table 3.** Evaluation results of the XP-Gaudan method

	GO term recognition (only)	GO term recognition + Association of GO terms with gene names
Precision	15.8%	8.7%
Recall	20.9%	15.8%
F-measure	18.0%	11.2%

We also applied the cross-products method and the Gaudan's method for the GO term recognition independently to the Biocreative IV corpus, in order to show the improvement of the XP-Gaudan method over them. Table 4 shows the evaluation results of the two individual methods, whose F-measures are lower than that of the XP-Gaudan method. Note that the recall of the XP-Gaudan method (21%) is close to the sum of the recall values of the two individual methods (26%), which may mean that the two methods target quite different sets of GO term occurrences. In other words, our proposed method based on the cross-products is quite well complementary to the Gaudan's method in detecting GO terms in text documents.

**Table 4.** Gaudan's method and analytical method using cross-product database

	Gaudan's method	Analytical method using cross-product database
Precision	17.2%	12.7%
Recall	12.1%	13.6%
F-measure	14.2%	13.1%

Our analytical method is based on the relative frequency of words that are among the top-K words of the concept. If the sum of relative frequencies of those words is greater than a threshold  $\theta$ , we regard the concept as expressed in the document. We tried different values of K and  $\theta$  to find optimal parameters for the method. The table 5 shows that the best value of K for the study is 10, while table 6 shows the best value of  $\theta$  is 100.

**Table 5.** XP-Gaudan method with different values of K

	K = 3	K = 5	K = 10	K = 15
Precision	20.7%	17.2%	15.8%	11.8%
Recall	10.0%	14.6%	20.9%	22.4%
F-measure	13.5%	15.8%	18.0%	15.5%

**Table 6.** XP-Gaudan method with different values of  $\theta$

	$\theta = 50$	$\theta = 100$	$\theta = 150$	$\theta = 200$
Precision	12.7%	15.8%	18.3%	20.2%
Recall	22.5%	20.9%	17.1%	13.8%
F-measure	16.2%	18.0%	17.7%	16.4%

## Conclusion

We presented a novel approach to GO term recognition, based on the cross-products of Gene Ontology and showed that it is complementary to a state-of-the-art method based on bag-of-words model. We will further develop sophisticated methods for gene name recognition and the association between GO terms and gene names.

## References

1. Camon, E.B., Barrell, D.G., Dimmer, E.C. et al. (2005) An evaluation of GO annotation retrieval for Biocreative and GOA. *BMC Bioinformatics*, 6(Suppl 1):S17.
2. Mungall, C.J., Bada, M., Berardini, T.Z., Deegan, J., Ireland, A., Harris, M.A., Hill, D.P., Lomax, J. (2011) Cross-product extensions of the Gene Ontology. *Journal of biomedical informatics* 44(1):80-86.
3. Gaudan, S., Jimeno Yepes, A., Lee, V., Rebholz-Schuhmann, D. (2008) Combining evidence, specificity, and proximity towards the normalization of gene ontology terms in text. *EURASIP journal on bioinformatics & systems biology*, page 342746.
4. Jonquet, C., Shah, N., Musen, M.A. (2009) The Open Biomedical Annotator. *Summit on Translational Bioinformatics*, pages 56–60.

5. Couto, F.M., Silva, M.J., Coutinho, P.M. (2005) Finding genomic ontology terms in text using evidence content. *BMC bioinformatics* 6(Suppl 1):S21.
6. Ruch, P. (2006) Automatic assignment of biomedical categories: toward a generic approach. *Bioinformatics* 22(6):658–64.
7. Rice, S.B., Nenadic, G., Stapley, B.J. (2005) Mining protein function from text using term-based support vector machines. *BMC Bioinformatics* 6(Suppl 1):S22.
8. Kim, J.J., Rebholz-Schuhmann, D. (2011) Improving the extraction of complex regulatory events from scientific text by using ontology-based inference. *Journal of Biomedical Semantics* 2(Suppl 5).
9. Kim, J.J., Luu, A.T. (2012) Hybrid pattern matching for complex ontology term recognition. In proceedings of the ACM Conference on Bioinformatics, Computational Biology and Biomedicine (ACM BCB), pages 289-296.
10. Kim, J.D., Ohta, T., Pyysalo, S., Kano, Y., Tsujii, J. (2009) Overview of BioNLP'09 shared task on event extraction. In Proceedings of the Workshop on BioNLP: Shared Task, pages 1–9.

# Gene Ontology Evidence Sentence Extraction and Concept Extraction: Two Rule-Based Approaches

Yu-De Chen<sup>1</sup>, Chia-Jung Yang<sup>1,2</sup>, Wen-Gan Li<sup>1</sup>, Chin-Yu Huang<sup>3</sup>, Jung-Hsien Chiang<sup>1,\*</sup>

<sup>1</sup>Department of Computer Science and Information Engineering, National Cheng-Kung University, Taiwan

<sup>2</sup>Department of Radiology, Mackay Memorial Hospital, Taitung Branch, Taiwan

<sup>3</sup>Department of Biology, University of Toronto, Canada

\*Corresponding author:

Tel: +886-6-275-7575 ext.62534, E-mail: [jchiang@mail.ncku.edu.tw](mailto:jchiang@mail.ncku.edu.tw)

## Abstract

Gene Ontology (GO) annotation have been relying on human annotation to capture accurate description of the published full-length literature. Though manual annotation may provide promising quality of the task. However, it is labour-intensive and time-consuming. In turn, we developed two different methods: a sequential pattern mining algorithm and GREPC (Geneontology concept Recognition- by Entity, Pattern, and Constrain) for the BioCreative GO track competition to recognize sentences that have mentioned functions or relevant information of genes and to catch the GO terms of these genes within them. In the results of subtask A, our best precision, recall, and F1-score were 0.212, 0.469, and 0.292, respectively; for subtask B, the best precision, recall, and F1-score were 0.150, 0.587, and 0.239, respectively. Our system based on GREPC had better evaluation scores, especially on recall rates in both of the subtasks.

## Introduction

GO is an uprising initiative within bioinformatics (1). The GO project serves to provide a bank of controlled vocabularies that is being applied to describe the gene products and their roles under three domains: biological process, molecular function, and cellular component (2). Up until now, GO annotation have been relying on human annotation to capture accurate description of the published full-length literature. Though manual annotation may provide promising quality of the task, with the ever-increasing volume of literatures being published, the labour-intensive and time-consuming disadvantages prevent researchers from continuing down this path. The alternative solution of this is therefore to apply computerized systems for GO curation and annotation. In turn, only a small portion of the data requires that of human effort to provide a training dataset. After feeding the system with that portion of training data, the remaining task will thereafter fall in the hands of the computerized systems. Some current

systems have shown promising results in determining a single domain of gene ontology annotation, such as within cellular component (3).

However, the BioCreative GO track competition seeks to resolute the limitations of computerized curation and to provide automated GO annotation in all three domains. The difficulties that have been encountered are mainly based on the complexities and variations that dwell within the writing styles of each individual author. Methods relying solely on Name Entity Recognition (NER) have shown to only capture the surface representations that are explicitly expressed by the authors. In other occasions, the ones that lie within the descriptions of their work may well require Natural Language Processing (NLP) and other semantic features of the sentences to be able to identify the relationships between the designated genes and their GO terms. Within Task IV of the BioCreative competition, we participated in two main subtasks: subtask A requires the retrieval of GO relevant sentences within the full-text articles; subtask B requires the identification of GO terms corresponding to their relevant genes. We aim to construct an integrated and automated system that can achieve this task to be as well as the results of human curations.

## **Material and Methods**

### **Gene Mention Identification**

We used the gene vocabulary data from NCBI (<ftp://ftp.ncbi.nlm.nih.gov/gene/>) and CTD (<http://ctdbase.org/downloads/>) to obtain all the possible synonyms of a specific gene ID (4, 5). For each given document-gene pair, we identified all the tokens started with or ended with this gene name, or its synonyms, in the document by case-insensitive regular-expression search. For example, we found “Anti-Baz, Baz, Bazooka, Par-3, UAS-Baz, anti-Baz, baz, bazXi106, bazXi106germ, bazooka, and par-3” to be the mentions of the gene baz (32703) within PMID: 10995441. We knew that anti-baz means the antibody of baz, but we did not exclude it. Bidirectional elongation of a gene name was not allowed because prefix and suffix of a gene were usually not present at the same time. The results of founded gene mentions for a document-gene pair were stored and used in all the runs of subtask A and subtask B.

### **The 1<sup>st</sup> and 2<sup>nd</sup> runs of subtask A and subtask B**

We developed a sequential pattern mining algorithm for mining frequency sequential patterns used in 1st and 2nd runs. The basic idea is similar to (6). The sequential pattern was accompanied with two classes. The first class was used to infer what GO term has appeared in a sentence. The other class was used to point out the GO term in which the gene within the same sentence belonged to. In short, each pattern was generated by the following steps: 1) preprocessing, 2) rule generation, 3) computing Support as well as Confidence, and 4) pruning.

### ***Preprocessing***

The preprocessing phase included several tasks: named entity recognition (NER), anonymization of genes, part-of-speech (POS) tagging, tokenization, and stemming. In the NER phase, we looked up gene mention list that was created in advance. Then, we anonymized the genes. For instance, a sentence "In vitro, CSC-1 binds directly to BIR-1" would become "In vitro, \_\_PROTEIN\_0\_\_ binds directly to \_\_PROTEIN\_1\_\_" so that we can further generate rules from the anonymized genes. A possible rule would look like "\_\_PROTEIN\_0\_\_ bind \_\_PROTEIN\_1\_\_". Their classes would be GO:0005515 (protein binding) and \_\_PROTEIN\_0\_\_, in which the former class meant the GO term it contained and the later represented the GO term in which the gene belonged to, called the owner of GO term. We can look up their original gene names through an index table created by the gene anonymization. We then tagged POS through the Stanford log-linear part-of-speech tagger. After POS tagging, each word, called a token, in a sentence would be tokenized through the identification of spaces between words. Finally, each token was stemmed by porter stemming algorithm.

### ***Rule Generation***

Each rule is a sequential pattern. The rules were learned from permuting token in a sentence rather than from all the tokens permuted from every sentence due to time and memory constraints. We also used the annotations from the corpus of BioCreative I as our training material. We set the sentence window size of 20 tokens to limit the permuted scope for avoiding massive calculation in a long sentence. By far, the generated rules were all candidate rules. They require yet a further step that examines their Support and Confidence.

*An example of the generated rule looks like the following:*

*prolifer\_NN, \_\_PROTEIN\_0\_\_, rate\_NN => GO:0008283, \_\_PROTEIN\_0\_\_  
,where NN stands for singular common nouns and GO:0008283 is the GO ID for cell proliferation.*

### ***Computing Support and Confidence***

Each of the generated rules was computed for their own Support and Confidence (7, 8). The difference to traditional association rule was that the order in the pattern was required. After computing Support and Confidence, we removed rules that had Support or Confidence lower than the thresholds. The threshold of Support can be used to filter out possible false rule and reduce the amount of calculation involved in rule generation. The threshold of Confidence can be used to measure how factual a rule is. For example, a rule, A1 & A2 => C, with Support 3 and Confidence 80%, we say the rule occurs 3 times in the training set and the probability of occurrence of the consequent, C, following an antecedent, A1 & A2, is 80%.

### ***Pruning***

For reducing the total number of rules, we removed a rule if it can be represented by their short form. For instance, both rules,  $A1 \ \& \ A2 \Rightarrow C$  with Support 3 and Confidence 80% and  $A1 \ \& \ A2 \ \& \ A3 = C$  with Support 3 and Confidence 80% appear in the rule set. The later rule would be replaced by the former because the short form can represent the long form due to identical Support and Confidence. After the Pruning phase, we can save a great quantity of memory space to retain other rules.

### ***Classification***

We used the generated rules to classify each of the sentences in the articles. The sentences were separated in the passages by LinPipe. Each of the sentences separated in the passages was compared to each of the rules. We used the consequents of the rule to classify sentences if the antecedents of the rule were matched perfectly in the order of the sentence. As we mentioned above, the form of consequents of rule is a GO term and an anonymized gene, e.g. GO:0003917, `__PROTEIN_0__`. Hence, we say the protein owns the GO term in the sentence if a rule can match the sentence.

### **The 3<sup>rd</sup> run of subtask A and subtask B**

Different from the 1st and 2nd runs in subtask A and B, we developed an independent system for the 3rd run with a core we called GREPC (Geneontology concept Recognition- by Entity, Pattern, and Constrain). In short, GREPC indexed the GO concepts based on three divisions: entity, pattern, and constrain. We gathered these kinds of information by text mining inside the GO database (1). Within that, we reconstructed the semi-structured name and synonyms for a GO concept into a better-structured synonym matrix. With GREPC, we can find GO terms in a sentence with a higher recall without losing much of the precision. For the 3rd run of subtask A and B, first, we split the given document into sentences. Second, we identified the sentences containing both the target gene and at least one GO term reported by GREPC. Third, we filtered out the sentences and GO terms made by the stop-entities. The stop-entities, which we picked from the results of the development dataset, were entities usually too general to be annotated, such as “protein”, “cell”, “kinase”, “DNA”, “antibody”, etc. Finally, we reported the non-overlapping sentences for subtask A and the non-repeated GO IDs for subtask B.

### **Results**

To evaluate our methods, we preformed the official latest evaluation system provided by the BioCreative organizer. The evaluation system can output three types of measurement values: recall, precision, and F1-score. Our system results are shown in Table 1 and Table 2. The highest scores in each run are in presented in boldface.

## Parameters

There are two parameters for sequential pattern mining: Support and Confidence. For the 1st run, we put negative instances into our training set to generate positive rules. In this case, we set the threshold of Support and Confidence to 2 and 0.8, respectively. For the 2nd run, we did not put negative instances into our training set to generate positive rules. We set 2 and 0.1 to Support and Confidence, respectively. The main difference between the 1st and the 2nd run is upon the intended aim for either the precision or the recall rate. For the 1st run, it applied strict conditions for rule generation so that the rules may have higher quality for inferring relevant sentences and GO terms. For the 2nd, on the other hand, in order to obtain a higher recall we allowed the rules to be generated more loosely so that we would have a greater quantity of rules, yet with lower quality.

**Table 1.** The results of evaluation of subtask A

	Run 1 <sup>st</sup>			Run 2 <sup>nd</sup>			Run 3 <sup>rd</sup>		
Parameter	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
0	0.205	0.011	0.020	0.110	0.351	0.167	<b>0.212</b>	<b>0.469</b>	<b>0.292</b>
1	0.114	0.006	0.011	0.053	0.171	0.081	0.138	<b>0.305</b>	<b>0.190</b>

**Table 2.** The results of evaluation of subtask B

	Run 1 <sup>st</sup>			Run 2 <sup>nd</sup>			Run 3 <sup>rd</sup>		
Parameter	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
0	<b>0.059</b>	0.006	0.011	0.008	0.068	0.015	0.037	<b>0.264</b>	<b>0.064</b>
1	<b>0.242</b>	0.020	0.037	0.034	0.336	0.062	0.150	<b>0.587</b>	<b>0.239</b>

## Post-Challenge Analysis

Using the gold standard passages from the developing and testing datasets of subtask A, we ran subtask B again with the GREPC. Since the gold standards contain the information of genes, the step of gene mention recognition was skipped. For a given document-gene pair, we restored the GO evidence sentences (GOES) according to the positional data in the gold standards. Then, we split the GOES into single sentences, and fed these sentences to GREPC. We also applied stop-entities filter to the output GO IDs of GREPC, just as we did in the contest. The results showed a great improvement of precision, a mild decrease of recall, and a significant increase of F1-score (Table 3). The results are reasonable because the gold standard GOES contain richer and better information than the original, whole document corpus. Without the gene mention recognition, we could avoid the errors happening in this step; and without dealing with the sentences outside the GOES, we could eliminate the false positive results from these sentences. Both this reasons helped us to have the better precision. The decrease of the recall implies that some of the correct

predictions of GO IDs were generated from the sentences outside the gold standard GOES. It was possible due to just coincidence, or missed annotations by human curators, or both. It requires manual check to prove or disprove these assumptions. Overall, our GREPC is robust to extract GO concepts from contexts, and the precision and recall rates are stable between developing and testing datasets.

**Table 3.** The results of post-challenge analysis. We ran the subtask B from the gold standard data from subtask A

	Developing dataset			Testing dataset		
Parameter	Precision	Recall	F-1	Precision	Recall	F1
0	0.098	0.193	0.130	0.092	0.231	0.132
1	0.345	0.421	0.379	0.306	0.458	0.367

## Discussion

During development of our first system, which was based on association rule, we encountered a serious problem of lacking training sentences. This system generated rules from the training data, and recognized GO concepts by these rules. Our system had poor ability to handle the GO terms that were never seen in the training corpus. Unfortunately, we found that most of the GO concepts in the developing dataset had not appeared in the training dataset. We supposed that the testing dataset would also have many novel GO concepts. In turn, we included the corpus from BioCreative I as one of the training sets, but the situation did not improve much (9). To overcome this shortage, we manually read the annotations of training dataset and tried to make better rules to cover the never-seen GO concepts. During this process, we noted that the information we needed were hidden in the GO database as a semi-structured form of names and synonyms, of GO concepts. Therefore, we used text mining techniques to extract these information inside the GO database, and built more than 63,000 rules, automatically. The newly-generated rules covered the whole spectrum of GO concepts, and had the ability to find the concepts deep down in the GO hierarchy. Because there was not enough time for delicate tuning, we turned in the system results as a backup in the 3rd run, just after the removal of the stop-entities. As in the post-challenge analysis, we knew the post-processing step, the stop-entity filter, was too primitive to get a higher precision. In the future, we can apply machine learning methods with features about the gene, the GO concept, the sentence structure, etc., to achieve a higher precision by removing the false positive results reported by GREPC.

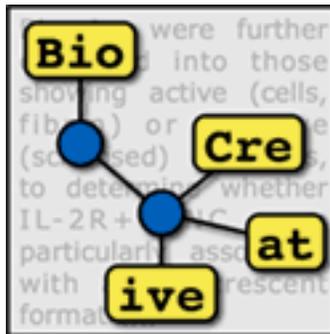
## Conclusion

For the subtasks of BioCreative GO tract, we developed two rule-based methods: one was based on association rule, mining the rules from the training and developing datasets; the other used

GREPC, extracting the rules from GO database. The later had a better performance possibly due to wider rule coverage of GO concepts. The method based on GREPC used almost no information from the training and developing datasets except for the stop-entities, which were manually picked from the results of the developing data, and surprisingly outperformed our former system. It demonstrated that the GREPC has a great potential of improvement by enrolling the abundant information from the corpus made by human curators.

## References

1. Michael Ashburner, Catherine A. Ball, and Judith A. Blake, et al. (2000) Gene ontology: tool for the unification of biology. *Nature Genetics*, **25,22** 25-9
2. Harris M.A., Clark J., and Ireland A., et al. (2004) The Gene Ontology (GO) database and informatics resource. *Nucleic Acids Research*, **32**
3. Kimberly Van Auken, Joshua Jaffery, and Juancarlos Chan, et al. (2009) Semi-automated curation of protein subcellular localization: a text mining-based approach to Gene Ontology (GO) Cellular Component curation. *BMC Bioinformatics*, **10**, **288**.
4. Donna Maglott, Jim Ostell, and Kim D. Pruitt, et al. (2005) Entrez Gene: gene-centered information at NCBI. *Nucl. Acids Res.*, **33**(suppl 1), D54-D58.
5. Davis AP, Murphy CG, and Johnson R, et al. (2013) The Comparative Toxicogenomics Database: update 2013. *Nucleic Acids Res.*, **41**(D1):D1104-14.
6. Bing Liu, Wynne Hsu, and Yiming Ma, (1998) Integrating Classification and Association Rule Mining. *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98)*, 80-86.
7. Rakesh Agrawal, Tomasz Imieliński, and Arun Swami, (1993) Mining Association Rules Between Sets of Items in Large Databases, *ACM SIGMOD Conference*, 207-216.
8. Rakesh Agrawal, and Ramakrishnan Srikant, (1994) Fast algorithms for mining association rules, *Proceedings of the 20th International Conference on Very Large Data Bases(VLDB)*, 487-499.
9. Lynette Hirschman, Alexander Yeh, and Christian Blaschke, et al. (2005) Overview of BioCreAtIvE: critical assessment of information extraction for biology. *BMC Bioinformatics*, **6**.



## TRACK 5 (IAT)

### Organizers:

- Cecilia N. Arighi, University of Delaware, USA
- Sherri Matis, AstraZeneca, USA
- Phoebe Roberts, Pfizer, USA
- Catalina O. Tudor, University of Delaware, USA

# BioCreative IV Interactive Task

Sherri Matis-Mitchell<sup>1#</sup>, Phoebe Roberts<sup>2#</sup>, Catalina O. Tudor<sup>3,4</sup> and Cecilia N. Arighi<sup>3,4\*</sup>

<sup>1</sup>Astrazeneca Pharmaceuticals, Wilmington, DE

<sup>2</sup>Pfizer, Boston, MA

<sup>3</sup>Center for Bioinformatics and Computational Biology, University of Delaware, Newark, DE

<sup>4</sup>Department of Computer and Information Sciences, University of Delaware, Newark, DE

\*Corresponding author: Tel: 302 831 3444, E-mail: arighi@dbi.udel.edu

#Contributed equally to this work

## Abstract

Fully automated text mining systems promote efficient literature searching, retrieval, and review but are not sufficient to produce ready-to-consume curated documents. These systems are not meant to replace curators, but they can assist in one or more biocuration steps. To do so, the interface with the curator is an important aspect that needs to be considered for tool adoption. The BioCreative Interactive task (IAT) is a track designed for exploring user-system interactions, promoting development of useful text mining tools, and providing a communication channel for the biocuration and the text mining communities. In BioCreative IV, nine text mining systems and around fifty curators participated in the IAT. Two levels of user participation with different commitment were offered to broaden curator involvement and obtain more feedback on usability aspects. The full level participation involved training on the system, curation of a set of documents with and without text mining assistance, tracking of time on task, and completion of the user survey. The partial level participation was designed to focus on usability aspects of the interface and not the performance *per se*. In this case, curators navigated the system by performing pre-defined tasks and they were asked whether they were able to achieve the task and how difficult it was to do it. In this manuscript, we describe many aspects of the development of the interactive task, from planning to execution, and discuss some preliminary findings for the systems tested.

## Introduction

BioCreative: Critical Assessment of Information Extraction in Biology is an international community-wide effort that evaluates text mining (TM) and information extraction (IE) systems applied to the biomedical domain (<http://www.biocreative.org/>) (1-5). A unique characteristic of this effort is its collaborative and interdisciplinary nature, as it brings together experts from various fields, including TM, biocuration, publishing houses and bioinformatics. Therefore, each competition is tailored towards specific needs of these communities. In particular, BioCreative has been working closely with biocurators to understand the various curation workflows, the text

mining tools that are being used and their major needs (6,7). In BioCreative Workshop 2012, descriptions of curation workflows from expert curated databases were reviewed to identify commonalities and differences among these workflows (7). Compared to a survey done in 2009, the 2012 results show that many more databases are now using text mining in parts of their curation workflows (6,7). Although text mining tools can be applied automatically to large corpora, these are not meant as replacement for biocuration, but rather to assist in this task. Therefore, interaction of the text mining tool with the user is a relevant aspect for its adoption.

To address the current barriers in using text mining in biocuration, BioCreative has been conducting user requirements analysis and user-based evaluations, and fostering standards development for text mining tool re-use and integration. In this respect, the BioCreative Interactive text mining Task (IAT) (8,9) has served as a means to observe the approaches, standards and functionalities used by state-of-the-art text mining systems with potential applications in the biocuration domain. The IAT task also provides a means for biocurators to be directly involved in the testing of text mining systems. The benefits to biocurators participating in this activity are multifold, including: direct communication and interaction with developers; exposure to new text mining tools that can be potentially adapted and integrated into the biocuration workflow, contribution to the development of text mining systems that meet the needs of the biocuration community, and dissemination of findings in peer-reviewed journal articles. A User Advisory Group (UAG) representing a diverse group of users with literature-based curation needs has been assisting in the design and assessment of the IAT. A subset of the UAG served as task coordinators (see author list), assuming a more active role in pre-testing systems, preparing surveys, and communicating with participants. This document describes the workflow of activities taking place in BioCreative IV (2013) and discusses some aspects based on preliminary findings.

## **Materials and Methods**

General information about the interactive task can be found in <http://www.biocreative.org/tasks/biocreative-iv/track-5-IAT>. Information for curators regarding the partial and full level evaluations, as well as links to the surveys are available at <http://www.biocreative.org/tasks/biocreative-iv/track-5-iat-activity-workflow>. For the usability test we reviewed and followed parts of the guidelines outlined here: <http://www.usability.gov>. The pre-designed evaluation tasks were presented to the user via SurveyMonkey (<https://www.surveymonkey.com/>) and responses were collected in CSV format. The user survey from BioCreative 2012 was modified to accommodate changes in scale and question wording to provide clearer choices to users, following the guidelines in the Questionnaire for User Interface Satisfaction (QUIS) developed by Chin et al. (see Figure 1 for examples of changes) (10). To facilitate analysis, the survey response options were converted from a semantic scale to a numerical scale of 1 to 5, indicating most negative to most positive feedback, respectively.

## Results and Discussion

One of our goals was to collect data from curators testing the systems, and provide useful feedback to developers of how to enhance or tailor the system for biocuration. For this, we conducted a usability test to evaluate each system. We tried to identify any usability problems, collect quantitative data on participants' performance (e.g., time on task, error rates), and determine participant's satisfaction with the system. It has been shown that usability testing can be done effectively in various settings including remote users (in different locations ) (<http://guidelines.usability.gov/guidelines/205>). This section describes various aspects of the development of the IAT, from planning to execution. The UAG (<http://www.biocreative.org/events/biocreative-iv/CFP/#committee>) has been engaged in defining various aspects of the task, including specifying the requirements for the systems, the reviewing of the user survey conducted in BioCreative 2012, recruiting curators, and testing the systems. **Figure 2** summarizes the general workflow. The steps of this workflow are explained below.

### Improvements of Biocreative IV Interactive Task (IAT) based on the previous IAT

In this section, we highlight what we learned from the previous IAT workshop and how we addressed these findings in BioCreative IV:

**1-Matching the system to the real world (of biocuration):** User interface language should cater to domain expert curators (as opposed to system developers or text mining experts). Moreover, it should follow standards of its users community (such as adopting controlled vocabularies, and be aware of their curation guidelines), and consider various levels of annotation (e.g., Sentence vs. Document level annotations).

*In BioCreative IV*, we asked the teams to engage two users in helping with the guidelines, and/or testing of their systems, to ensure the teams' understanding of the biocuration community that they were targeting.

**2-Testing the systems NOT the users:** At no point were the participants tested for this task. However, in the context of this activity, we needed to distinguish between novice and expert curators, as this could have an impact on reporting the performance of the systems.

*In BioCreative IV*, we asked, via a survey, for the years of experience as a biocurator in order to aid later analysis of task outcomes. In addition, we explicitly mentioned that the users were not being tested.

**3-Providing extensive documentation:** To ensure a positive feedback from the users, the system should come with detailed annotation guidelines, as well as a tutorial of how to use the system, with hands-on examples. Guidelines should provide examples on both what and what not to annotate.

*In BioCreative IV*, we requested these guidelines and tutorials, and also reviewed them before making them available to the curators.

**4-Emphasizing system performance and interface functionalities:** We conclude from previous IATs that adding functionalities to assist in the curation task can increase efficiency of experienced curators even when the system performance is not optimal. Therefore both must be in place to make a responsive system.

*In BioCreative IV*, we emphasized the importance of useful functionalities right in the call for participation, and provided examples of interesting features incorporated in existing interfaces.

**5-Specifying system output:** To be useful for the curation task, annotated results should be made available in standard formats that can be further utilized in the curation workflow.

*In BioCreative IV*, we asked, where applicable, that results could be downloaded in BioC format (13). We expect to distribute the annotated corpora that were generated in this format. In addition, a human readable format such as CSV was also requested.

### **Interactive Task Phases** (*Figure 2*)

#### *1-Specifications for the task*

To come up with the specifications for the text mining systems participating in the interactive task, we collected a list of general features and functionalities from previous interactive tasks and asked the User Advisory Group (UAG) to assist in reviewing and prioritizing them. For this, we conducted a short activity which consisted of performing some basic curation tasks in two systems, namely GeneView (11) and PubTator (12), which covered majority of the functionalities in the list. The exercise was meant to inspire UAG members to think about the requirements in a practical way, as opposed to involving them in an abstract discussion about systems requirements. After the exercise, the UAG members were asked to select the four most important items from the list. **Table 1** shows the votes from 11 participants.

We classified the requirements into three different categories: mandatory, strongly desired, and nice-to-have. The four most important items were (1) highlighting entities and relationships, (2) processing full text, (3) editing text mining results, and (4) the ability to export the results in standard formats. We agreed to keep “process full text” as a *highly desired* feature instead of *mandatory*, as we recognize that not all curation workflows require full text (e.g., some databases use only the abstract section in the triaging step). As supporting material, we prepared a document including real examples of those requirements. This information was added to the description of the interactive task sent during the call for participation (URL:<http://www.biocreative.org/tasks/biocreative-iv/track-5-IAT/>).

**Table 1**-Prioritization of system requirements. Each UAG member selected the 4 most important functionalities. The table shows the votes from 11 members.

<b>Functionalities</b>	<b>#Votes</b>
Highlight entities and relationships	8
Process full text	8
Allow manual mode for annotation*	7
Ability to edit results*	5
Export curated results in standard formats	5
Sort the results according to different criteria; rank the results based on what is more relevant to the user.	3
Interactive disambiguation of domain entities	3
Display curation suggestions or warnings for display during curation	2
Report wrong, ambiguous or missing synonyms	2
Upload a gene list	1

\* Although both of these functionalities reflect the ability to edit text mining results, they were viewed as distinct entities by UAG members.

## *2-User Survey*

In last year's interactive task, we prepared a survey to capture the user's experience with the system (9). The survey consisted of five main categories, namely: overall reaction, system's ability to help complete the tasks, design of application, learning to use the application, and usability. For BioCreative IV, after discussion with the UAG, the survey was reviewed and improved in language, as well as in scaling the options. The survey simplifies the rating scales, and provides more opportunities to comment. **Figure 1** shows a comparison of a set of questions used in previous IAT tasks versus the current survey. We also asked the curator to provide information about their years of experience with the task in order to distinguish novices from experts. The complete new survey is available at <http://ir.cis.udel.edu/biocreative/survey2.html> and completion was mandatory for curators who participated at the full curation level.

### BioCreative 2012

System's ability to help complete tasks		1	2	3	4	5	6	7	NA
8. I am able to accomplish tasks quickly using this system:	disagree	<input type="radio"/>	agree <input type="radio"/>						
9. I am able to accomplish tasks effectively using this system: (i.e., the system helps me get closer to my curation goal)	disagree	<input type="radio"/>	agree <input type="radio"/>						
10. I am able to accomplish tasks efficiently using this system: (i.e., with this system I can be both fast and effective)	disagree	<input type="radio"/>	agree <input type="radio"/>						

### BioCreative IV (2013)

System's ability to help complete tasks		
<p>Speed: the system decreases the time it takes to reach my curation goal:</p> <p><input type="radio"/> strongly disagree</p> <p><input type="radio"/> disagree</p> <p><input type="radio"/> neutral</p> <p><input type="radio"/> agree</p> <p><input type="radio"/> strongly agree</p> <p><input type="radio"/> NA</p> <p>comments?</p> <input type="text"/>	<p>Effectiveness: the system helps me get closer to my curation goal:</p> <p><input type="radio"/> strongly disagree</p> <p><input type="radio"/> disagree</p> <p><input type="radio"/> neutral</p> <p><input type="radio"/> agree</p> <p><input type="radio"/> strongly agree</p> <p><input type="radio"/> NA</p> <p>comments?</p> <input type="text"/>	<p>Efficiency: with this system I can be both fast and effective:</p> <p><input type="radio"/> strongly disagree</p> <p><input type="radio"/> disagree</p> <p><input type="radio"/> neutral</p> <p><input type="radio"/> agree</p> <p><input type="radio"/> strongly agree</p> <p><input type="radio"/> NA</p> <p>comments?</p> <input type="text"/>

**Figure 1-** Snapshot of questions within the “System’s ability to help complete tasks” category in Biocreative 2012 (top) and 2013 (bottom) user surveys.

### 3-Pre-defined tasks and survey

We asked users to perform pre-defined short tasks to allow them to navigate the system and provide initial feedback on first impressions about the system. Some examples of pre-defined tasks were:

- log in/install software required for curation
- test functions that lead up to curation (e.g. search, upload, sort and/or rank)
- require system to perform an erroneous task
- edit a text mining result
- save results
- describe the meaning of some icons/buttons/tabs

We followed with questions about their ability to perform the task and how difficult it was to accomplish. In some cases, we also looked into navigation cues in the user interface by presenting a snapshot of the interface and asking about meaning of specific menu options or buttons. In the end, all surveys contained general questions about the usability of the system applicable to all systems. The surveys were reviewed by multiple coordinators to ensure that they contained clear directives that could be reproduced.

The link to the activity for each system is available at

<http://www.biocreative.org/tasks/biocreative-iv/track-5-iat-activity-workflow/#Partial>

### Team participation (Figure 2.2)

Similar to the previous Interactive Tasks, we invited text mining teams to submit a document describing the system and the proposed biocuration task(s), provide the URL to a functioning

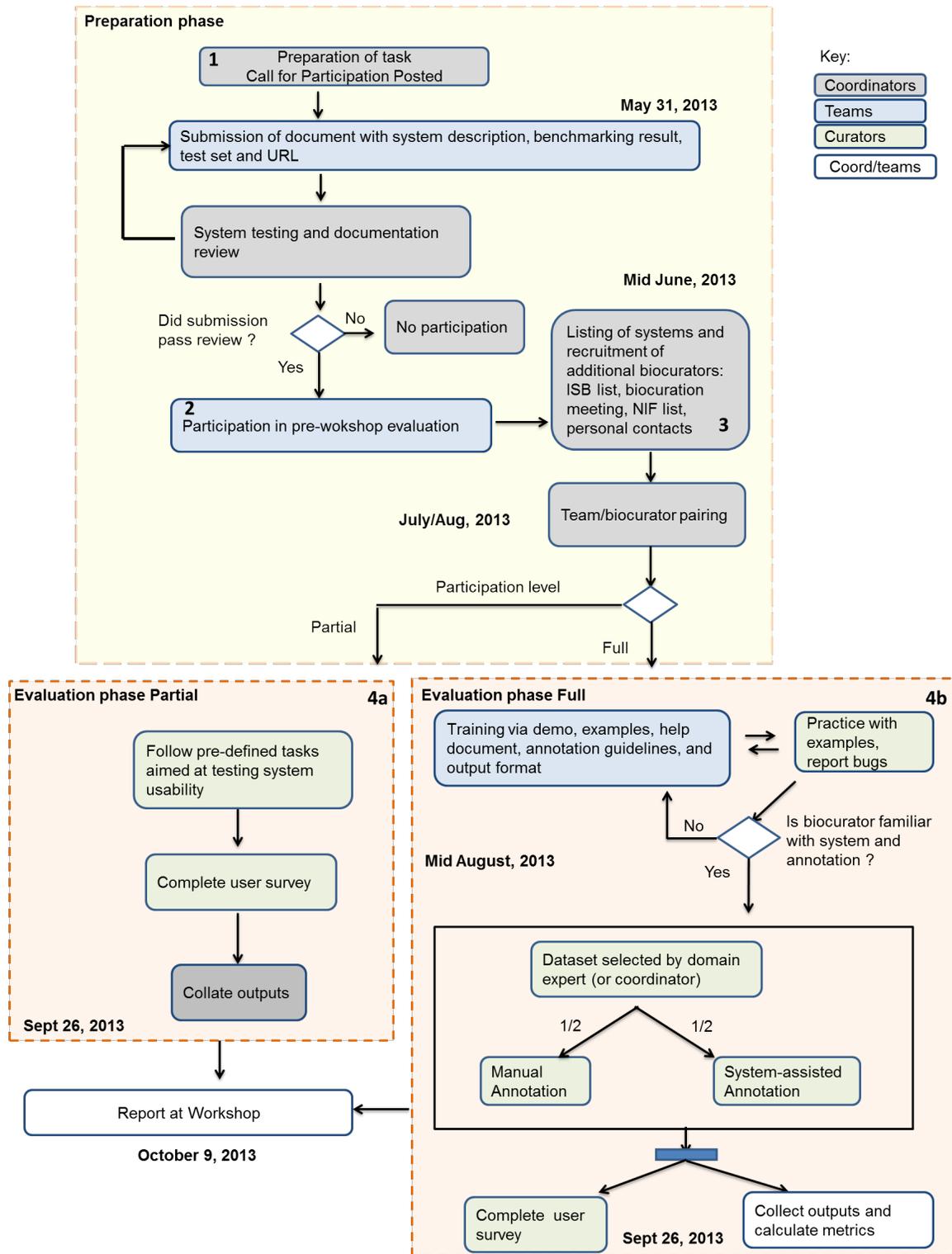
system, and address aspects such as relevance and impact (explain user and target community), user interactivity (provide interface with requirements), and performance (if the system has undergone a prior evaluation). In addition, each team was asked to engage two users from its user community to assist with the corpus selection, the curation guidelines, and the testing.

The organizers of the task (coordinators) reviewed the systems with the help of two other UAG members to make sure that systems complied with the specifications and had a realistic plan to finish the implementation by the time of testing. The coordinators spent a significant amount of time making sure that the systems were working and that they provided the necessary guidelines (both curation and system-related) and tutorials for the users.

Nine teams participated in the Interactive Task. This represents an increase from six systems that fully participated in the last BioCreative IAT workshop. **Tables 2 and 3** describe the participating systems. They differed in the complexity of their interface. Some offered workflow design options (Argo), management systems for curation (Egas), on-the-fly training (tagtog), active learning (tagtog, SciKnowMine), and plug-ins for the web browser (MarkerRIF). Others provided more traditional interfaces that offered common input and output of text mining results with highlighting and sorting/scoring capabilities (CellFinder, BioQRator, RLIMS-P, Ontogene). These systems also differed in the type of tasks that they performed (information retrieval and/or entity recognition and/or event recognition). Despite the highly desired requirement of full text processing, only three tools utilized full text articles (Ontogene, SciKnowMine and tagtog). In a few cases, systems could support full text processing (EGAS, Argo, and RLIMS-P), but this was not pursued in the BioCreative task due to time constraints, either from the implementation or the curation side.

### **Curator recruitment (Figure 2.3)**

We reached out to the biocuration community by presenting the activity in a BioCreative workshop hosted at the International Biocuration meeting in Cambridge, UK (April, 2013). We also submitted the call for participation via the International Biocuration Society and the Neuroscience Informatics Framework mailing lists, via UAG members, and personal contacts. Given the busy schedule of biocurators, we offered full and partial levels of participation, which involved different levels of commitment from the user. The full level participation involved curation of a defined corpus with and without text mining assistance, tracking the respective time on task, and completing the user survey. The partial level participation involved performing pre-defined tasks as described earlier.



**Figure 2-** Interactive task track workflow. See “Interactive Task Phases” section for descriptions of each numbered phase.

**Table 2-**Description of systems that participated in the Interactive Task in BioCreative IV

System	Description of the tool	URL	Browser compatibility*			
			Fx	Ch	Sf	IE
<b>Cell Finder</b>	Annotation of gene, expression relation and cell type in text snippets from a set of articles	<a href="http://141.20.31.85/cellfinder/">http://141.20.31.85/cellfinder/</a>	x	x	x	x
<b>Ontogene</b>	Detection of Gene/Chemical/Diseases and their interactions	<a href="http://marvin.cl.uzh.ch/kitt/bcms/bc2013-ctd/">marvin.cl.uzh.ch/kitt/bcms/bc2013-ctd/</a>	x		x	
<b>MarkerRIF</b>	Retrieval of articles about biomarkers, and extraction of disease and biomarker (gene) with normalization	<a href="http://bws.iis.sinica.edu.tw/MarkerRIF">bws.iis.sinica.edu.tw/MarkerRIF</a>	x	x		
<b>SciKnowMine</b>	Triage based on pre-trained categories of interest in full length articles	<a href="http://www.isi.edu/projects/sciknowmine/triage_system_demo">http://www.isi.edu/projects/sciknowmine/triage_system_demo</a>	x	x		x
<b>BioQRator</b>	Retrieval based on relevance on protein-protein interaction information and annotation of protein pair	<a href="http://www.bioqrator.org/">http://www.bioqrator.org/</a>	x	x	x	x
<b>RLIMS-P</b>	Triage on protein phosphorylation. Annotation of kinase, substrate and site with normalization.	<a href="http://research.bioinformatics.udel.edu/text_mining/rlimsp2/">http://research.bioinformatics.udel.edu/text_mining/rlimsp2/</a>	x	x	x	
<b>Egas</b>	Identification and extraction of protein-protein interaction events described over PubMed abstracts related to neuropathological disorders	<a href="http://bioinformatics.ua.pt/egas">bioinformatics.ua.pt/egas</a>	x	x	x	
<b>tagtog</b>	Annotation of gene names within full-text documents especially machine-predicted documents	<a href="http://www.tagtog.net">www.tagtog.net</a>	x	x		
<b>Argo</b>	Annotation of metabolic process-related named entities, namely chemical entities and genes or gene products	<a href="http://argo.nactem.ac.uk">argo.nactem.ac.uk</a>	x	x	x	

\*Fx=Firefox, Ch=Chrome, Sf=Safari and IE=Internet Explorer

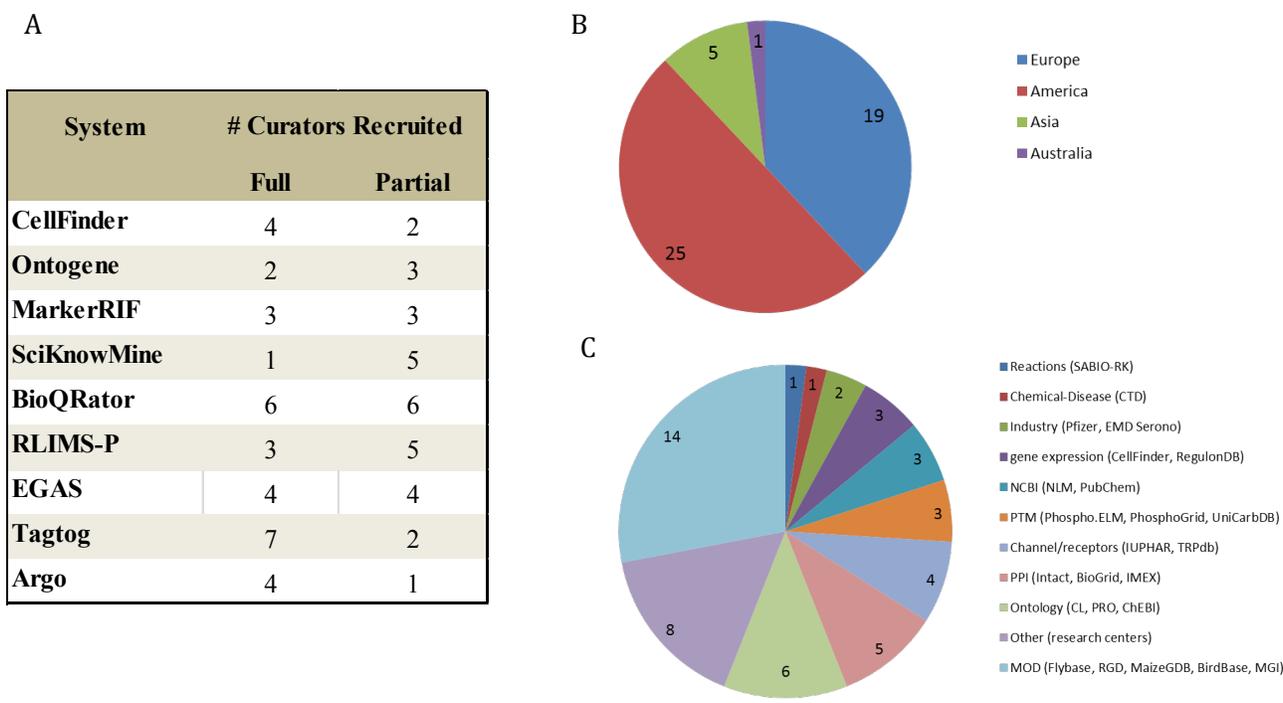
A total of 50 curators participated in the interactive task in different capacities (as of October 4). **Figure 3** shows the distribution by system and level of participation. All systems were inspected by at least 5 curators.

### Evaluation (Figure 2.4)

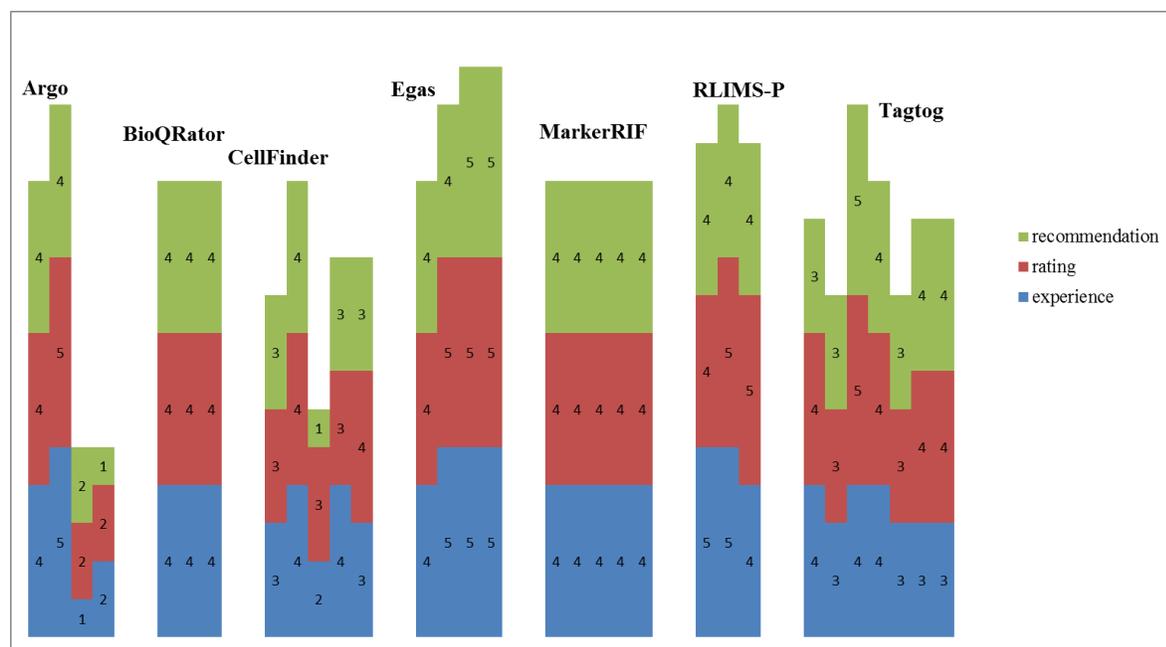
Similar to IAT 2012, we evaluated the systems in terms of performance and preference. Performance metrics include: precision, recall, F-measure, and efficiency of curation (e.g., the change in time when using the text mining tool vs. other basic methods). The preference metrics include reported satisfaction with the system while performing predefined tasks, and post-testing satisfaction.

**Table 3-** Text mining tasks performed by systems and other highlights

System	Triage/Retrieval	Entity Recognition	Event Recognition	Entity Normalization	Text	Relevant aspects
<b>Cellfinder</b>		gene name tissue/anatomy part	expression event		text snippets from full-text or abstracts	The text mining tool is separately run by developers and results presented to users. Only highlighted information in snippets can be edited
<b>Ontogene</b>		gene name chemical names disease gene environmental growth conditions (EGC)	Gene-Chemical-Disease interactions EGC-gene association	EntrezGene CTD vocabulary CTD vocabulary	full-text XML	Tool can be adapted for different curation workflows (see RegulonDB)
<b>MarkerRIF</b>	Disease-biomarker	gene name	disease-biomarker	EntrezGene	abstract	It is a browser add-on and allows highlight and annotation in PubMed environment
<b>SciKnowMine</b>	Mouse phenotype				full-text PDF	Needs software installation, but it is open source. Training and classification of corpus on the fly but run via command lines.
<b>BioQRator</b>	Protein-Protein interaction	protein name		EntrezGene	abstract	Queries are PubMed style. Can build your corpus. Results are ranked according to PIE algorithm.
<b>RLIMS-P</b>	Phosphorylation	protein kinase phosphorylated protein phosphorylated site	phosphorylation event	UniProtKB UniProtKB	abstract (full-text tested but to be implemented)	Queries are PubMed style. Results can be grouped based on kinase or substrate
<b>EGAS</b>		protein name	protein-protein interaction event		abstract (full-text can be processed)	Collaborative platform for curation, with manager and users. Allow both manual or text mining modes
<b>Tagtog</b>		gene names			full-text XML	Provides training on-the-fly. Custom-build dictionaries can be uploaded
<b>Argo</b>		chemical protein disease	metabolic process- related action words	ChEBI UniProtKB CTD	abstract (full-text can be processed)	Custom workflow design. Allows both manual and text mining modes



**Figure 3.** Distribution of curators (A) by system and participation level, (B) by continent, and (C) by type of hiring institution. A total of 50 curators participated in this activity. Notice that the total number in (A) is higher because some curators tested more than one system.



**Figure 4-** Stack bar graph showing overall satisfaction measure for each system. Each line represents a curator involved in the full level participation. Overall satisfaction includes the rating of the user experience (blue), the rating of the system (red), and the recommendation of the system (green). The rates range from 1 to 5, from the most negative to the most positive feedback.

### **Preliminary analysis of results**

**Figure 4** shows the overall satisfaction of the systems per curator (only the systems with feedback from more than 3 curators are included). The satisfaction is based on the full level evaluation task. Majority of satisfaction ratings were consistent and positive (i.e., with a score greater than 3) among curators evaluating each system. An exception is Argo where we observe disagreement between the first and the second halves of the curators. Evaluating this system was challenging as this was the most complex system that participated in the track. The low rating in satisfaction seems to be related to the learnability curve and the organization aspect of the system. Nevertheless, we would like to point out that Argo is flexible and a powerful tool which allows the building of workflows by combining various modules.

We are still collecting information regarding the time spent on the task with vs. without the text mining tool's assistance. However, we can comment on a couple of the systems. For Egas, the curation time was significantly reduced for 3 of the 4 curators by 1.5-4 times. It is noteworthy that the increase in efficiency was solely due to the text mining component (the same interface was used in both assisted and manual modes). In the case of RLIMS-P, the curators provided positive overall impression of the system, but the time spent on curation in manual vs. system assisted mode was almost equal for 2 of the 3 curators. The survey and follow-up with curators indicated that the positive satisfaction level was due to the ability of the tool to find abstracts highly relevant to phosphorylation, as well as the functionalities included online, which helped the curators find quickly and highlight information in the interface. Both curators expressed that the bottleneck was the gene normalization step, which is not the focus of this system. RLIMS-P is planning to improve this step once it moves to full length articles.

Based on the partial level testing, we identified some areas of improvement for some of the systems. One is concerned with the error messages: there were cases where curators could not understand why no error messages or un-satisfactory ones were displayed when no documents were retrieved for non-existing PMIDs (e.g., Ontogene, BioQRator, RLIMS-P). Another area of improvement is with regard to key functionalities that were not found by the user (e.g., classification of articles in SciKnowMine). Thirdly, the color choice was not optimal for color blinded users in the case of Egas, for example. Lastly, some of the icons and names of sections/functionalities were hard to understand (e.g., in RLIMS-P, the eye icon is intended for articles that were viewed but not annotated, or in BioQRator, it was not clear how the PIE score for sorting the data was calculated).

We provided each team with all the feedback from the users. Updates on the evaluation for all systems will be presented in the workshop.

## **Conclusion**

The interactive task provides the opportunity for text mining and biocuration communities to closely interact. This activity is highly challenging to organize. Part of the challenge comes from IAT not having a common task with a common data set and scoring metrics. A typical shared task is optimal for comparing and evaluating TM algorithms in batch mode. However, IAT aims to involve fully developed and complex systems. Thus, making it a shared task would limit the participation, as undue effort would be required from developers and curators for tasks they may not fully appreciate. Synchronizing the schedule for so many systems and curators is also challenging for this task.

Perhaps one of the most difficult challenges we have in this task is the recruitment of curators, who tend to be skeptical of text mining systems. This is usually a consequence of previous involvement in such activities, where creation of gold standard corpora for NLP competitions did not provide immediate optimal solutions. Moreover, curators tend to have high expectations of the tools. This may discourage participation of some NLP teams who do not believe that their systems are ready for curators need. Our goal is to break these barriers, by encouraging and improving communications between curators and system developers, and by involving curators in a more active role in the development of text mining systems. The increasing number of participants in this task, as well as the cooperation between curators and the teams, showed that we are moving a step closer to accomplishing our goal.

## **Funding**

This work was supported by the National Science Foundation [ABI-1062520 to C.N.A. and C.O.T.] and the National Library of Medicine of the National Institutes of Health [G08LM010720 to C.N.A. and C.O.T.]. The content is solely the responsibility of the authors and does not necessarily represent the official views of the funding organizations.

## **Acknowledgements**

We would like to thank Cathy Wu who first introduced the IAT in BioCreative, and also the BioCreative Organizing committee for their support. We are grateful to Andrew Chatr-aryamontri and Sandra Orchard for assisting in the system's pre-testing, and all other UAG members: Judith Blake, Stan Laulederkind, Donghui Li, Fiona McCarthy, Peter McQuilton, Mary Schaeffer, and Kimberly Van Auken, for their continuous support for the BioCreative activities. We would like to thank Ben Carterette (University of Delaware) and Sangya Pundir (EBI) for their suggestions on the usability test and surveys. Last but not least, a special thanks to all the teams and curators for their effort and participation in this activity.

**Conflict of Interest:** none declared.

## References

- 1.Hirschman, L., Yeh, A., Blaschke, C. and Valencia, A. (2005) Overview of BioCreative IV: critical assessment of information extraction for biology. *BMC Bioinformatics*, **6**, S1.
- 2.Krallinger, M., Morgan, A., Smith, L., Leitner, F., Tanabe, L., Wilbur, J., Hirschman, L. and Valencia, A. (2008) Evaluation of text-mining systems for biology: overview of the Second BioCreative community challenge. *Genome Biology*, **9**, S1.
- 3.Leitner, F., Mardis, S., Krallinger, M., Cesareni, G., Hirschman, L. and Valencia, A. (2010) An Overview of BioCreative II.5. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **7**, 385 - 399.
- 4.Arighi, C., Lu, Z., Krallinger, M., Cohen, K., Wilbur, W., Valencia, A., Hirschman, L. and Wu, C. (2011) Overview of the BioCreative III Workshop. *BMC Bioinformatics*, **12**, S1.
- 5.Wu, C.H., Arighi, C.N., Cohen, K.B., Hirschman, L., Krallinger Martin, Lu, Z., Mattingly, C., Valencia, A., Wieggers, T.C. and Wilbur, W.J. (2012) Editorial: BioCreative-2012 Virtual Issue. *Database (Oxford)*.
- 6.Hirschman, L., Burns, G.A., Krallinger, M., Arighi, C., Cohen, K.B., Valencia, A., Wu, C.H., Chatr-Aryamontri, A., Dowell, K.G., Huala, E. *et al.* (2012) Text mining for the biocuration workflow. *Database (Oxford)*, **2012**, bas020.
- 7.Lu, Z. and Hirschman, L. (2012) Biocuration workflows and text mining: overview of the BioCreative 2012 Workshop Track II. *Database*, **17**.
- 8.Arighi, C., Roberts, P., Agarwal, S., Bhattacharya, S., Cesareni, G., Chatr-aryamontri, A., Clematide, S., Gaudet, P., Giglio, M., Harrow, I. *et al.* (2011) BioCreative III interactive task: an overview. *BMC Bioinformatics*, **12**, S4.
- 9.Arighi, C. and Ben Carterette, K.B.C., Martin Krallinger, W. John Wilbur, Petra Fey, Robert Dodson, Laurel Cooper, Ceri E. Van Slyke, Wasila Dahdul, Paula Mabee, Donghui Li, Bethany Harris, Marc Gillespie, Silvia Jimenez, Phoebe Roberts, Lisa Matthews, Kevin Becker, Harold Drabkin, Susan Bello, Luana Licata, Andrew Chatr-aryamontri, Mary L. Schaeffer, Julie Park, Melissa Haendel, Kimberly Van Auken, Yuling Li, Juancarlos Chan, Hans-Michael Muller, Hong M Cui, James P. Balhoff, Johnny Chi-Yang Wu, Zhiyong Lu, Chih-Hsuan Wei, Catalina O. Tudor, Kalpana Raja, Suresh Subramani, Jeyakumar Natarajan, Juan Miguel Cejuela, Pratibha Dubey, Cathy Wu. (2012) An Overview of the BioCreative 2012 Workshop Track III: Interactive Text Mining Task. *Database (Oxford)*.
- 10.Chin, J.P., Diehl, V.A. and Norman, K.L. (1988) Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface. *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*, 213-218.
- 11.Thomas, P., Starlinger, J., Vowinkel, A., Arzt, S. and Leser, U. (2012) GeneView: a comprehensive semantic search engine for PubMed. *Nucleic Acids Res*, **40**, 12.
- 12.Wei, C.-H., Kao, H.-Y. and Lu, Z. (2012) PubTator: A PubMed-like interactive curation system for document triage and literature curation. *Proceedings of the 2012 BioCreative Workshop (Washington DC, USA)*, 152-157.
- 13.Comeau, D.C., Islamaj Dogan, R., Ciccarese, P., Cohen, K.B., Krallinger, M., Leitner, F., Lu, Z., Peng, Y., Rinaldi, F., Torii, M. *et al.* (2013) BioC: a minimalist approach to interoperability for biomedical text processing. *Database*, **18**.

# Evaluation of the CellFinder pipeline in the BioCreative IV User Interactive task

Mariana Neves<sup>1,2</sup>, Julian Braun<sup>2</sup>, Alexander Diehl<sup>3</sup>, G. Thomas Hayman<sup>4</sup>, Shur-Jen Wang<sup>4</sup>, Ulf Leser<sup>1</sup>, and Andreas Kurtz<sup>2,5</sup>

<sup>1</sup> Humboldt-Universität zu Berlin, Knowledge Management in Bioinformatics, Berlin, Germany

<sup>2</sup> Berlin Brandenburg Center for Regenerative Therapies, Charité, Berlin, Germany

<sup>3</sup> Department of Neurology, University at Buffalo School of Medicine and Biomedical Sciences, Buffalo, USA

<sup>4</sup> Rat Genome Database, Medical College of Wisconsin, Milwaukee, USA

<sup>5</sup> Seoul National University, College of Veterinary Medicine, Research Institute Veterinary Science, Seoul, Korea

## Abstract

We present results on the participation of the CellFinder text mining pipeline for curation of gene/protein expression in anatomical parts in the BioCreative IV User Interactive task. The pipeline integrates state-of-the-art and freely available tools for the following steps: triage of potentially relevant documents, retrieval of documents, preprocessing, named-entity recognition, event extraction and a graphical user interface for manual validation of the results. Four curators have been recruited for this evaluation and have suggested three topics of interest: kidney-related diseases in rat, human dendritic cells and human mesenchymal stem cells. Each curator validated gene/protein expression events automatically extracted from 30 Medline abstracts. A total of 634 expression events were obtained from the three datasets and approximately 35% of them (216 events) were validated as being correct, which is a level of precision slightly lower than previous experiments with internal CellFinder curators.

## Introduction

Biomedical literature curation is the process of automatically and/or manually compiling biological data from scientific publications and making it available in a structured and comprehensive way. This task requires careful reading of publications by domain experts, which is known to be a time-consuming task.

The BioCreative IV User Interactive task (IAT)<sup>1</sup> is a community-driven task which aims to bring together biocurators and developers of text mining solutions. Participant teams are required to present a Web-based system for a biocuration task of their choice. External biocurators recruited by the organizers can choose any of the available tools (and biocuration tasks) and be engaged in

---

<sup>1</sup> <http://www.biocreative.org/tasks/biocreative-iv/track-5-IAT/>

hands-on experiments by validating a small set of documents. Tools are evaluated regarding their usability and the accuracy of the automatic predictions.

For the BioCreative IV User Interactive task (IAT), we have participated with a text mining pipeline which has been developed in the scope of the CellFinder database<sup>2</sup>. The task consisted of curating gene/protein expression events in cell types, tissues and organs and the pipeline has been previously evaluated for curation for kidney-related cells [1]. In addition to the validation of the information automatically extracted by the text mining pipeline, curators were asked to manually annotate the gene/protein expression events present in the documents to allow comparison between manual and text mining-supported curation. The sentence below illustrates an example of protein expression in cells (PMID 18989465):

On the other hand, the *podoplanin expression* occurs in the differentiating *odontoblasts* and the expression is sustained in differentiated odontoblasts, indicating that odontoblasts have the strong ability to express podoplanin.

Four external curators have agreed to participate in the validation of the CellFinder pipeline. Two of them belong to the Rat Genome Database<sup>3</sup> and are experts in gene and disease curation. One of them has a PhD in microbiology with over 25 years of experience in molecular genetics in numerous model organisms and has spent the last four years engaged in gene, disease, phenotype and pathway curation for rat, human and mouse. The other has a PhD in developmental biology with a dissertation on embryonic blood vessel formation. She has research experience in cancer biology, biomedical engineering and stroke research. The third curator has six years of experience in mesenchymal stem cell (MSC) research and during his PhD investigated how *ex vivo* MSC progenitors adapt to *in vitro* conditions. Finally, the fourth curator has a Ph.D. in molecular cell biology, seven years of postdoctoral and biotech experience in experimental immunology and genomics, and ten years of experience in biocuration. Each curator proposed a topic of interest, which were: kidney-related diseases in rat, human dendritic cells and human mesenchymal stem cells.

In the next sections we present an overview of the CellFinder text mining pipeline, details on the processing of the document collections and preliminary results of this experiment.

## Materials and Methods

For the BioCreative IV IAT task, we have proposed the CellFinder text mining pipeline for curation of gene/protein expression events in cells, tissues and organs. The four external curators proposed three topics, which are listed below along with the corresponding code which we will cite throughout this work when referring to each of these datasets.

---

<sup>2</sup> <http://cellfinder.org/>

<sup>3</sup> <http://rgd.mcw.edu/>

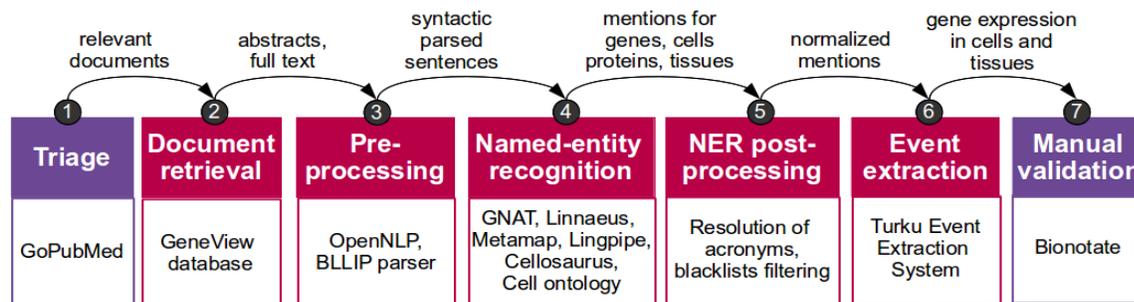
Kidney: rat renal-related diseases, such as end stage renal disease and chronic renal insufficiency;

Mesenchymal: human mesenchymal stem cells;

Dendritic: human dendritic cells.

## Text Mining Pipeline

The CellFinder curation pipeline includes the following steps: triage of relevant documents, retrieval of full text, linguistic pre-processing, named-entity recognition (NER), post-processing, gene expression extraction and manual validation of the results (cf. Figure 1). No adaptation or retraining of the pipeline was carried out for any of the proposed topics, except on the keywords provided in the triage step. Each of these steps is briefly described below; more details can be found in Neves et al. [1].



**Figure 1.** CellFinder text mining pipeline for the BioCreative IV IAT task. Automatic procedures are shown in red and manual ones in purple.

### *Triage (1) and Document Retrieval (2)*

Since curators were asked to validate and annotate only a small set of 30 abstracts, we utilized GoPubMed [2] for the retrieval of relevant documents. The three queries we used were the following: Rats[mesh] "gene expression"[go] "Kidney Failure, Chronic"[mesh] (Kidney dataset), "Mesenchymal Stem Cells"[mesh] Humans[mesh] "Gene Expression"[mesh] (Mesenchymal dataset) and "Dendritic Cells"[mesh] Humans[mesh] "Gene Expression"[mesh] (Dendritic dataset). Provided with the list of PMIDs exported from GoPubMed, we retrieved the abstracts from the database developed in the scope of the GeneView tool [3].

### *Pre-processing (3)*

Documents were first split by sentences using the OpenNLP toolkit<sup>4</sup> and then parsed using the BLLIP parser<sup>5</sup> [4] (also known as the McClosky-Charniak parser). Part-of-speech tags, tokenization and full parsing were derived from the BLLIP parser output.

<sup>4</sup> <http://opennlp.apache.org/>

### ***Named-entity Recognition (4) and NER Post-processing (5)***

Named-entity recognition was performed for the following types: genes/proteins, cell lines, cell types, anatomical parts and expression triggers. Triggers were extracted based on a list of 509 terms which was built manually and matched to the text using Lingpipe<sup>6</sup>. We identified genes using GNAT [5], a system for extraction and normalization of gene/protein mentions. Cell lines were recognized based on version 6.31 of Cellosaurus<sup>7</sup>, a manually curated vocabulary of cell lines. Matching to the text was carried out with Linnaeus [6]. For the recognition of cell types and anatomical parts, we used Metamap [7], a system for UMLS (Unified Medical Language System) concept extraction. Cell types have also been extracted using an ontology-based approach in which synonyms from the Cell Ontology (CL) are matched against the text using Linnaeus [6].

Regarding the post-processing step, we included an extra acronym resolution for cell types, besides the one carried out by the Metamap tool. Additionally, we used a list of potential false positives for gene/protein, cell, organ and tissue name which was initially created based on common errors performed by NER tools and was recently updated with feedback received from the kidney curation experiment [1].

### ***Event Extraction (6)***

Gene/protein expression events were automatically extracted using the Turku Event Extraction System (TEES) [8]. It was trained on 10 full texts on human embryonic stem cells which have been manually annotated and whose evaluation was presented in [1]. Each gene/protein expression event is always composed of a gene/protein and a cell line, cell type or anatomical part (tissue, organ).

### ***Manual Validation (7)***

Manual validation of the automatically predicted gene/protein events was carried out based on Bionotate [9], a tool designed to support collaborative curation of biomedical data. We have configured Bionotate for curation of gene/protein expression data as shown in Figure 2. Bionotate loaded one snippet, or text segment at a time, which was randomly selected from the repository of extracted events. Each snippet was highlighted with only one gene/protein expression event composed of three entities: one expression trigger, one gene/protein and one cell line, cell type, tissue or organ. For each predicted event, we presented a snippet of text containing the sentence where the events were supposed to be taking place along with the preceding and following two sentences. Additionally, Bionotate presented the identifier of the document from which the data came, along with a link to PubMed, buttons for removing and

---

<sup>5</sup> <https://github.com/dmcc/bllip-parser>

<sup>6</sup> <http://alias-i.com/lingpipe/>

<sup>7</sup> [ftp://ftp.nextprot.org/pub/current\\_release/controlled\\_vocabularies/cellosaurus.txt](ftp://ftp.nextprot.org/pub/current_release/controlled_vocabularies/cellosaurus.txt)

adding new annotations, and a question that assesses the curation task (e.g., gene/protein expression) with a list of possible answers.

From the list of relevant PMID identifiers returned by GoPubMed (cf. Triage), the 60 top documents of the Kidney dataset were randomly split into two groups of 30 abstracts: Kidney1 and Kidney2. The snippets derived from the 30 abstracts in each of the Kidney1, Kidney2, Mesenchymal and Dendritic datasets were loaded into Bionotate and the URL was sent to the corresponding curator. They were asked to check whether the entities had been correctly extracted and whether a gene/protein expression event was described in the text. Curators were free to change the span of the entities, as long as the entity has been at least partially automatically annotated. For example, in Figure 2 the curator could change the gene/protein annotation (in blue) from “TLR” to “TLR4” or even to “Toll-like receptor 4”, which is a synonym, but should not change it to another gene/protein, such as “MMP-2”. Thus, changes to the entities’ span should only be carried out in the context of the “Entities of interest” listed above the snippet.

Finally, one of the answers had to be chosen to assess the text mining results with respect to event extraction, named-entity recognition and document triage. Event extraction was assessed by answers 1, 2 and 3. Answer 1 was selected when all entities were correctly identified (whether automatically or after corrections) and if they were indeed taking part in a gene/protein expression event. This was also true when the text described low expression of a gene/protein, as it still constituted a gene/protein expression. Answer 2 was also only to be selected if all entities were correctly identified and were taking part in a gene expression event. The difference from Answer 1 is that in order to select Answer 2, the text must indicate that there was negation, in other words, the gene/protein was not being expressed in the specified anatomical part. Thus, data derived from answers 1 and 2 were potential candidates to be integrated into the CellFinder database, as a positive and negative expression level, respectively. Finally, Answer 3 was only important as feedback for the event extraction component of the text mining pipeline. It should be chosen if all entities were correctly identified (whether automatically or after corrections) but there was no gene/protein expression event taking place, i.e., both entities were being cited in some other context.

Answers 4, 5, 6, 7 were important as feedback for the name-entity recognition components of the text mining pipeline and indicated whether the expression trigger, gene/protein, cell/anatomy or both of them, respectively, were incorrectly extracted (not even partially). Finally, Answer 8 was important as feedback for the document triage component of the text mining pipeline. This option could be selected when the text seemed not to be related to cell research, to the topic which was suggested or to the characterization (gene/protein expression) of cells and anatomical parts.

After choosing one of the options above and clicking on the “save annotation” button, an XML file was generated for each snippet and saved in the server with the changes in the entities (if any) and the chosen answer. A new snippet was then loaded on screen and the validation process continued.

### Snippet

Id: 23041150\_sent\_000002\_000006\_context\_000295\_000967\_Expression\_T52\_Cell\_T45\_Gene\_T56

Extracted from article: PubMed [23041150](https://pubmed.ncbi.nlm.nih.gov/23041150/)

Entities of interest:

Expression : expression

Cell Type : MSCs

Gene : TLR4

Astragaloside IV (AS-IV) was widely used for the treatment of cardiovascular diseases in China. The aim of this study was to determine the effect of AS-IV on bone marrow mesenchymal stem cells (MSCs) and the underlying mechanism in diabetes. We used reverse transcription polymerase chain reaction and western blotting to determine the expression of Toll-like receptor 4 (TLR4), matrix metalloproteinase-2 (MMP-2) and NF-κB p65 in MSCs under high glucose (HG) with or without pretreatment with AS-IV. The surface expression of TLR4 was checked by flow cytometry and the expression of TNF-α and MCP-1 were detected by ELISA in diabetes patients treated with AS-IV. AS-IV promoted the proliferation of MSCs and attenuated the increased expression of TLR4 induced by HG.

Expression: expression	x
Gene: TLR4	x
Cell Type: MSCs	x

Mark selected text as:

Does this snippet support a gene expression between the provided gene and cell line or cell type?

- 1. Yes, an event is taking place and all entities are correct.
- 2. Yes, but the text says the gene expression is NOT taking place.
- 3. No, no event is taking place although all entities are correct.
- 4. No, this is no gene expression trigger.
- 5. No, this is no gene.
- 6. No, this is no cell or anatomical part.
- 7. No, both gene and cell or anatomical part are incorrect.
- 8. No, the snippet (publication) seems to be irrelevant for CellFinder.

**Figure 2:** Screenshot of Bionotate for the mesenchymal stem cell dataset. For this example, the first answer would be selected as a gene expression event is indeed taking place in the sentence which contains the highlighted entities.

## Manual Curation

For each of the three topics proposed by the external curators, a manual annotation of 30 abstracts was carried out by the curator who suggested the topic. This was the same set of documents which was processed by the text mining pipeline, in order to allow a comparison of manual and text mining annotation. However, different curators were asked to carry out the manual annotation and text mining validation of the same set of documents, allowing computation of inter-annotator agreements. Manual annotation of the abstracts was supported by the use of the Brat annotation tool<sup>8</sup>.

<sup>8</sup> <http://brat.nlplab.org/>

## Results and Discussion

As of the submission deadline for this manuscript, manual annotation of the abstracts was still ongoing. Therefore, we present here only the results for the processing and validation of the datasets using the CellFinder text mining pipeline. Additionally, curator 3 agreed to carry out an additional validation of the automatic predictions from the Mesenchymal dataset to enable computation of inter-annotator agreement.

The results derived from the automatic processing of the 120 abstracts are shown in Table 1. Although each dataset contained exactly 30 abstracts, the size of the documents varied considerably, as demonstrated by the total number of sentences, which differed by more than 100 between the Kidney1 and Dendritic datasets. All collections contained a fairly high number of gene/proteins, tissues/organs and triggers words, but only a few cell line mentions.

Statistics/Datasets	Kidney1	Kidney2	Dendritic	Mesenchymal
<b>no. documents</b>	30	30	30	30
<b>no. docs with events</b>	25 (83%)	21 (70%)	16 (53%)	26 (87%)
<b>no. sentences</b>	407	394	289	327
<b>no. genes/proteins</b>	393	439	308	340
<b>no. cell lines</b>	36	14	39	53
<b>no. cell types</b>	72	57	184	230
<b>no. tissues/organs</b>	502	474	230	467
<b>no. triggers</b>	465	481	362	401
<b>no. gene expression events</b>	108	119	187	220

**Table 1:** Statistics on the annotations which have been automatically compiled from the four datasets.

The only large difference among the datasets is the small number of cell type annotations for the two Kidney collections, in contrast to the Dendritic and Mesenchymal sets. This might be due to two reasons: (i) the collection contained many irrelevant documents, and/or (ii) the text mining pipeline recall for rat cell types was rather low. However, the percentage of documents in which a gene/protein expression was found was not much lower when compared to the other datasets, which means that most of the documents indeed contained gene/expression events. Future comparison between manually annotated abstracts and text mining processed ones might also shed additional light on the recall of cell type predictions.

Table 2 shows the statistics on the eight answers from the validation using Bionotate of the gene/protein expression events extracted by the text mining pipeline. The answers can be

summarized as follows. Almost 35% (answers 1 and 2) of the gene expression events have been extracted correctly, as well as the participating entities. This included both positive and negative statements of gene expression in cells and anatomical parts. This is a lower value than the approximately 52% precision which was previously measured during internal curation of the CellFinder database on a large dataset of more than 2,000 full texts [1]. Around 26% (answers 3 and 4) of the snippets described processes not related to gene expression, although the gene, cell and anatomy were correctly recognized, as opposed to 17% reported in the previous evaluation. Finally, 38% (answers 5, 6 and 7) of the extracted events contained a wrongly identified gene/protein, cell/anatomy or both of them, which is larger than the 25% previously reported.

Answers/Datasets	Kidney1	Kidney2	Dendritic	Mesenchymal	Total
<b>1. Gene expression</b>	40 (37.1%)	44 (37.0%)	61 (32.6%)	71 (32.3%)	216 (34.1%)
<b>2. Neg. gene exp.</b>	-	-	5 (2.7%)	-	5 (0.8%)
<b>3. No gene exp.</b>	8 (7.4%)	21 (17.6%)	66 (35.3%)	30 (13.6%)	125 (19.7%)
<b>4. Wrong trigger</b>	6 (5.6%)	5 (4.2%)	2 (1.1%)	25 (11.4%)	38 (6.0%)
<b>5. Wrong gene</b>	17 (15.7%)	15 (12.6%)	39 (20.9%)	32 (14.5%)	103 (16.2%)
<b>6. Wrong cell/anat.</b>	32 (29.6%)	25 (21.0%)	7 (3.7%)	52 (23.6%)	116 (18.3%)
<b>7. Wrong entities</b>	5 (4.6%)	9 (7.6%)	6 (3.2%)	2 (1.0%)	22 (3.5%)
<b>8. Irrelevant doc.</b>	-	-	1 (0.5%)	8 (3.6%)	9 (1.4%)
<b>Total</b>	<b>108 (100 %)</b>	<b>119 (100%)</b>	<b>187 (100%)</b>	<b>220 (100%)</b>	<b>634 (100%)</b>

**Table 2:** Evaluation of the gene expression snippets in Bionotate.

When comparing results across the four datasets, the percentage of correct gene/protein expression events was similar and ranged from 32% to 37%. However, the percentage of incorrect extracted events (no expression event despite having correct entities) was much higher in the Dendritic dataset (around 35%) in contrast to the Kidney and Mesenchymal datasets (7% to 17%). An analysis of 10 of the 66 snippets classified with Answer 3 for the Dendritic dataset showed that some of the answers provided by the curator were correct but that some snippets could have been classified as incorrect cell type or gene/protein instead.

Regarding the recognition of the named-entities, again there was some difference between the Dendritic and the other datasets. Gene/protein was classified 20% of the times as incorrect for the Dendritic dataset while only 12%-15% of the time for the Kidney and Mesenchymal datasets. Wrongly extracted gene/proteins included acronyms, such as “SCC” (squamous-cell carcinoma), cell types, such as “dendritic cell”, and anatomy-related terms, such as “pancreatic”. On the other hand, precision of cell and anatomical parts extraction was excellent for the Dendritic dataset (4% incorrect) and good for the other sets (21%-29% incorrect). Indeed, for the Kidney and Mesenchymal datasets, false positives for cells and anatomical parts included mentions such as

“down”, “analyzed”, “time”, “stem” and “poly”. However, this answer was also mistakenly assigned to correct annotations, such as “extracellular matrix”, “kidney”, “macrophage”, “plasma”, “bone” and “leukocyte”.

Inter-annotator agreement was assessed for the Mesenchymal dataset using the results provided by curators 3 and 4. Both curators provided the same answer for 47% of the snippets. Differences occurred mainly when distinguishing between Answer 3 and 4, which have similar meanings, and for answers related to mistakes derived from the named-entity recognition step. The rather low agreement rate shows both the difficulty of the task and possibly also some deficiencies in the curation guidelines. In spite of this, the good percentage of correct gene/protein expression events (32%-37%) across three distinct topics demonstrates the suitability of the text mining pipeline to the proposed task.

## Funding

This work was supported by Deutsche Forschungsgemeinschaft [grant numbers KU 851/3-1, LE 1428/3-1 to AK and UL], and European Commission [grant number 334502 to AK]. GTH and S-JW were supported by a grant (HL64541) from the National Heart, Lung and Blood Institute on behalf of the National Institutes of Health.

## Acknowledgments

We are thankful to Philippe Thomas for support in document retrieval from the GeneView.

## References

1. Neves, M., Damaschun, A., Mah, N., et al. (2013) Preliminary evaluation of the cellfinder literature curation pipeline for gene expression in kidney cells and anatomical parts. *Database*.
2. Doms, A. & Schroeder, M. (2005) GoPubMed: exploring PubMed with the gene ontology. *Nucleic Acids Res.* 33, W783–W786.
3. Thomas, P., Starlinger, J., Vowinkel, A., Arzt, S. & Leser, U. (2012) Geneview: a comprehensive semantic search engine for PubMed. *Nucleic Acids Res.* 40, W585–W591.
4. Charniak, E. & Johnson, M. (2005) Coarse-to-fine n-best parsing and maxent discriminative reranking. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, ACL '05, 173–180 (Association for Computational Linguistics, Stroudsburg, PA, USA).
5. Hakenberg, J., Plake, C., Leaman, R., Schroeder, M. & Gonzalez, G. (2008) Inter-species normalization of gene mentions with gnat. *Bioinformatics* 24, i126–i132.
6. Gerner, M., Nenadic, G. & Bergman, C. (2010) Linnaeus: A species name identification system for biomedical literature. *BMC Bioinformatics* 11, 85.
7. Aronson, A. R. & Lang, F.-M. (2010) An overview of MetaMap: historical perspective and recent advances. *J. Am. Med. Inform. Assoc.* 17, 229–236.
8. Bjerne, J., Ginter, F. & Salakoski, T. (2012) University of Turku in the BioNLP'11 shared task. *BMC Bioinformatics* 13, S4.

9. Cano, C., Monaghan, T., Blanco, A., Wall, D. P. & Peshkin, L. (2009) Collaborative text-annotation resource for disease-centered relation extraction from biomedical text. *J Biomed Inform* 42, 967–977.

# Assisted curation of growth conditions that affect gene expression in *E. coli* K-12

Socorro Gama-Castro<sup>1</sup>, Fabio Rinaldi<sup>2</sup>, Alejandra López-Fuentes<sup>1</sup>, Yalbi Itzel Balderas-Martínez<sup>1</sup>, Simon Clematide<sup>2</sup>, Tilia Renate Ellendorff<sup>2</sup>, Julio Collado-Vides<sup>1</sup>

<sup>1</sup> Centro de Ciencias Genómicas, Cuernavaca, UNAM, México.

<sup>2</sup> Institute of Computational Linguistics, University of Zurich

## Introduction

RegulonDB [4] is a database with manually curated knowledge extracted from the literature describing knowledge of transcriptional regulation in *E. coli* K-12. It contains objects such as genes, promoters, transcription factor binding sites (TFBSs), transcription factors (TFs), terminators and operons. It contains relations among those objects, such as regulatory interactions among TFs and genes, promoters and operons. An important piece of information for the adequate description of knowledge on gene regulation is that of growth conditions (GC) and their corresponding control conditions (CC), which are used in experiments to identify regulatory interactions. Currently RegulonDB has only a small set of GCs, which are known to activate or repress the transcription of a few genes. A list of the mechanisms by means of which the GCs affect gene expression is still missing.

The process of curation of the GC would require keeping track of a large amount of data about the experiment, such as the name of the GC, the control of the experiment, the growth media used, the temperature, the pH, the type of effect (induction or repression) provoked by the transition from CC to GC to the regulated gene, the TF and sigma factor involved, when known, in such regulatory mechanism. Thus, the biocuration challenge we face is to extract this type of relevant information from the large corpus of around 5,000 papers, supporting the knowledge of mechanisms present in RegulonDB with experiments performed since the 80s or even 70s to date. Doing this work manually would involve a considerable amount of time. This challenge motivated us to initiate our collaboration with experts in text mining tools, and use resources such as OntoGene/ODIN, to simplify and as such accelerate our curation.

The goal of this project is to verify which GCs activate or inhibit the transcription of the genes of *E. coli*, as well as to identify the type of mechanism used. In a first instance we can determine the type of mechanism based upon the identification of the TF and the effect it causes on some of the regulated genes under the given GC. Therefore we will try to identify the name of the experimental condition, the affected gene, the type of effect, and the TF involved in such regulatory process.

OntoGene/ODIN provides a flexible, customizable environment for document-centric curation approaches. The OntoGene team at the University of Zurich, working in collaboration with RegulonDB curators, adapted ODIN to the specific needs of this project. Ontogene/ODIN has been previously described in several publications [1-3]. In the rest of this short paper we describe the results of the experiment on curation of GC for RegulonDB using ODIN.

## Methods

We used the complete list of genes of *E. coli* from RegulonDB for building dictionaries to be used by OntoGene/ODIN. Additionally RegulonDB provides words that indicate the type of effect caused under a given GC (*activation*, *repression* and a complete list of their synonyms). Our initial work has been performed on a set of 46 articles from RegulonDB that were selected because of their connection with the genes related to the regulon of OxyR, and with the regulatory interactions, operons, promoters and terminators of those genes. The articles have been automatically annotated by the OntoGene pipeline using the terminology provided by RegulonDB, which includes types such as GENE, EFFECT, Transcription Factors (TF), etc. We use the sentence filters of ODIN to visualize, in those 46 articles, only those sentences containing the name of a GENE and a word of type EFFECT. Since we know that OxyR is a TF which is involved in the regulation of genes which respond to oxidative stress, we expect to find relevant data about GC in that set of articles.

Since we have only an incomplete list of GC we cannot use the elements of the list as a filtering criteria to select relevant sentences in ODIN, since such a choice would severely limit the results. Our goal is in fact to discover the possible names and synonyms for GC. Because of that reason, after applying the filter, we use the automatically annotated genes and effects, but we manually mark previously missing GCs, in order to generate an extended set of such conditions.

## Results

There were 36 out of the 46 articles with at least one sentence containing GCs-related information, which show the effect of a GC on the expression of at least one gene (see example in figure 1). Of these 36 articles, 20 contain at least one sentence that describes the mechanism of regulation at work under the specified GC. See an example in figure 2.

### Figure 1.

S189 Based on the results of the 2 - D separation and the data obtained from the transcriptional fusion analysis , it is clear that the **viaK** - S operon is **induced** in the presence of **L - ascorbate** .

### Figure 2.

S40 DNA binding activity of **ArcA** were first reported in studies of **sodA** , encoding the manganese - containing superoxide dismutase , which is **repressed** by the **arcA** gene product during **anaerobic growth** ( 5 , 45 ) .

Other types of sentences found are those containing only information about the regulation of a gene by a TF (Regulatory Interaction), without mentioning the GC.

These results show that ODIN is a very useful instrument to help in the manual curation of RegulonDB. Some observations made during this experiment will help generate improved versions of the tools and terminological resources. For example, since all TFs are also genes, they received a duplicate annotation. However, if we want to curate only GC-related sentences, it would be better not to include the TFs in the list of genes used as a filter, because the terms related to an EFFECT, which are found in a sentence with GC, are also found in the sentences that contain only information of regulatory interactions. We also encountered GC-related data that regulate the activity of the TFs, and their mechanism, even if not necessarily at the level of transcription. This is also useful information for RegulonDB.

We spotted some errors in the automated annotation of some gene names. In particular short words (typically 4 letters or less) might also happen to be gene names. For example, “fold” is frequently used to express the level of expression of a gene, rather than to refer to the gene of the same name. Such errors were manually corrected.

A long-term goal is to use the system for a more specific, accurate and efficient curation. In the process of the experiment described above, we realized that it would be useful to be able to distinguish interrogative or hypothetical sentences from affirmative ones, since only the latter provide reliable data for curation. Another problem to solve is the lack of clarity when a mutant is mentioned in a sentence, without a description in the same sentence, since the sentence-based curation approach hides the information needed for complete understanding (being it contained in a non-selected sentence). A similar problem is caused by anaphoric mentions such as “this gene”, where the actual name of the gene is mentioned in a previous sentence, which might not be shown when the filter is active.

As soon as we have a complete list of GC, we will be able to use the ODIN sentence filters to allow a very detailed inspection of the documents and obtain more specific results.

An additional result of this practical experiment was to collect different ways in which GCs terms are described in articles. For example, stress conditions with hydrogen peroxide and exponential phase are written in different phrases with different words, such as:

- *H2O2, H2O2 exposure, H2O2 stress, H2O2 treatment, H2O2 - stressed cells, hydrogen peroxide, presence of hydrogen peroxide, exposure to hydrogen peroxide, hydrogen peroxide treatment, high concentrations of hydrogen peroxide, treated with hydrogen peroxide, treatment with hydrogen peroxide*
- *during growth, exponentially growing, exponential growth, exponential phase, exponentially growing cells, logarithmic – phase, log phase*

### **Analysis of the paper: PMID 21908668**

The article with PMID 21908668 was analyzed in detail. The sentence splitter currently used in the OntoGene system identifies 841 sentences in this paper, although without the references there are only 327 of them. When the “GENE and EFFECT” filter was applied, 78 sentences were selected (three of them are part of the references).

From the total of 78 sentences: 18 sentences describe the regulation of a TF on a gene (TF-gene regulation) under a specific condition (TF-gene-GC); three sentences describe TF-gene regulation as well as the regulation of a GC on a gene (GC-gene regulation), although generally it is the same gene in both cases, it is not shown clearly the dependence of the GC with the TF. 13 sentences describe only GC-gene regulation. 12 sentences describe only TF-gene regulation. Five sentences are not clear, because they are questions, and not affirmative or negative sentences; 13 sentences contain general data about TF, e.g., it could be mentioned that a such TF is regulating a set of genes, but it is not specified to which genes; and finally 14 sentences do not contain the expected data, for example this group could have another kind of regulation where a TF or a GC is not involved.

In summary:

18	TF-gene- GC
3	TF-gene + GC-gene
13	GC-gene
12	TF-gene
5	Confusing sentences
13	TF
14	Nothing

The sentences that we expect to find belong to the first four groups, they represent 65% of the sentences when they are filtered. We also have the option to eliminate the TF group (with 13 sentences) if we exclude from the list of genes the names of the TFs. This would change the relevant set of sentences from 65 to 78%.

### **Conclusion and future work**

The experiment clearly shows that efficient text mining tools coupled with a customized interface can significantly increase the efficiency and productivity of specific biocuration activities. The activity described in this paper required about 20 hours for the curation of 46 papers. Since the normal curation process at RegulonDB requires about four hours per paper, it can be estimated that the same activity, without the support of OntoGene/ODIN, would have required about 184 hours. It seems therefore that the careful introduction of sophisticated text mining and curation tools can improve the efficiency of curation nearly 10-fold.

We intend to continue the activities described in this paper within the scope of a planned collaborative project with curators of RegulonDB group and with the support of the OntoGene team. The goal is to gradually automatize much of the most tedious activities of the curation process, and therefore free up the creative resources of the curators for more challenging tasks, and enabling a much more efficient curation process.

## References

1. Fabio Rinaldi, Thomas Kappeler, Kaarel Kaljurand, Gerold Schneider, Manfred Klenner, Simon Clematide, Michael Hess, Jean-Marc von Allmen, Pierre Parisot, Martin Romacker, Therese Vachon (2008). OntoGene in BioCreative II. *Genome Biology*, 2008, 9:S13, PMC2559984
2. Fabio Rinaldi, Simon Clematide, Yael Garten, Michelle Whirl-Carrillo, Li Gong, Joan M. Hebert, Katrin Sangkuhl, Caroline F. Thorn, Teri E. Klein, Russ B. Altman. *Using ODIN for a PharmGKB revalidation experiment. The Journal of Biological Databases and Curation*, Oxford Journals, 2012, bas021; doi:10.1093/database/bas021
3. Fabio Rinaldi and Simon Clematide and Simon Hafner and Gerold Schneider and Gintare Grigonyte and Martin Romacker and Therese Vachon. Using the OntoGene pipeline for the triage task of BioCreative 2012. *The Journal of Biological Databases and Curation*, Oxford Journals, 2013. doi:10.1093/database/bas053
4. Salgado H, Peralta-Gil M, Gama-Castro S, Santos-Zavaleta A, Muñiz-Rascado L, García-Sotelo JS, Weiss V, Solano-Lira H, Martínez-Flores I, Medina-Rivera A, Salgado-Osorio G, Alquicira-Hernández S, Alquicira-Hernández K, López-Fuentes A, Porrón-Sotelo L, Huerta AM, Bonavides-Martínez C, Balderas-Martínez YI, Pannier L, Olvera M, Labastida A, Jiménez-Jacinto V, Vega-Alvarado L, Del Moral-Chávez V, Hernández-Alvarez A, Morett E, Collado-Vides J. RegulonDB v8.0: omics data sets, evolutionary conservation, regulatory phrases, cross-validated gold standards and more. *Nucleic Acids Res.* 2013 Jan;41(Database issue):D203-13. doi: 10.1093/nar/gks1201. Epub 2012 Nov 29.

# ODIN: a customizable literature curation tool

Fabio Rinaldi<sup>1</sup>, Allan Peter Davis<sup>2</sup>, Christopher Southan<sup>3</sup>, Simon Clematide<sup>1</sup>, Tilia Renate Ellendorff<sup>1</sup>, Gerold Schneider<sup>1</sup>

<sup>1</sup> Institute of Computational Linguistics, University of Zurich

<sup>2</sup> Department of Biology, North Carolina State University, Raleigh, NC 27695-7617, USA

<sup>3</sup> IUPHAR Database and Guide to PHARMACOLOGY web portal. The University British Heart Foundation Centre for Cardiovascular Science. The Queen's Medical Research Institute. University of Edinburgh, Edinburgh EH16 4TJ, United Kingdom

## Introduction

ODIN is a lightweight graphical interface for literature curation that can be run within a web browser. ODIN has been developed by the OntoGene group (<http://www.ontogene.org/>) at the University of Zurich, which specializes in biomedical text mining, in particular extraction of domain entities and their relationships from the scientific literature. The quality of their text-mining technologies has been evaluated several times through participation in community-organized competitive evaluation challenges, where OntoGene frequently obtained top-ranked results [2].

Currently ODIN is coupled with the OntoGene pipeline, which provides its text mining capabilities; however, nothing prevents ODIN from being interfaced with other text-mining services, as long as they support the same data exchange format. In order to achieve optimal performance and user satisfaction, the OntoGene team typically customizes the OntoGene pipeline and ODIN for the specific curation task. OntoGene and ODIN have already been customized for some experiments in assisted curation in collaboration with well-known databases, in particular PharmGKB, CTD and RegulonDB, which have been described in a number of journal publications [3].

As part of their participation in the triage task (task 1) of BioCreative 2012 [4], the OntoGene team produced a version of OntoGene/ODIN for the Comparative Toxicogenomics Database (CTD; <http://ctdbase.org/>) [6]. Customization involves using the entities of interest for the specific database, and minor adaptations of the interface to better suit the specific needs of the curators (*e.g.*, providing links from annotated entities to the web pages of the reference database).

While the OntoGene team obtained the best overall results in the official evaluation, this was concerned only with the capability of the underlying text mining system to deliver entities of

interest for CTD (genes, chemicals, and diseases) and articles ranked according to their relevance for CTD curation, so ODIN was not part of the evaluation.

In February 2013, the OntoGene team initiated a collaboration with the RegulonDB group (<http://regulondb.ccg.unam.mx/>) [5], which aims at improving their curation efforts through adoption of state-of-the-art text-mining techniques and advanced curation interfaces. As part of this collaboration the two groups decided on a joint participation in the interactive task (task 3) of BioCreative 2013. ODIN was, therefore, gradually modified according to suggestions provided by RegulonDB.

In August 2013, the organizers of the shared task required access to the curation interface in order to allow external curators to experiment with it. Since at this point the customization for RegulonDB was not yet completed, the OntoGene team decided to make available for this purpose the CTD version of ODIN, which in the meantime had been extended and already included several of the new features developed for RegulonDB.

As a result of these circumstances, two slightly different versions of ODIN have been evaluated in BioCreative 2013: ODIN-RegulonDB (described in a separate paper [1]) and ODIN-CTD. In the rest of this paper we will briefly describe the latter and the results of the independent evaluation.

## **Methods and Results**

ODIN-CTD (like every version of ODIN) is available as a web application (HTML + ExtJS) and can be used from any browser. However due to incomplete support of web standards by some browser vendors, the OntoGene team recommends to use Firefox, Safari or Chrome (in this order).

Three curators invited by the task organizers were given access to ODIN-CTD (one of them chose to remain anonymous). All of them used firefox. At the first access the user is prompted to enter a login identifier that he/she can freely chose as long as it used consistently at later access of the system (the anonymous user pointed out that it was not clear that the identifier was not pre-assigned). The login identifier is stored as a cookie by the browser and therefore the user will not be prompted for it again as long as the same browser and machine are used. There is, however, an option to change the identifier if needed.

While we provide an extensive user manual, the experiments showed that this might not be a very effective way to explain how to use the tool given the limited time that curators have to perform the assigned tasks. Therefore, at a later stage in the evaluation, we added a series of screencasts that describe in a simple fashion the main functionalities of the system.

After login, the users have the option to either inspect one of the sample files provided by the system, or process an arbitrary PubMed abstract by entering the corresponding PubMed ID. The abstract will then be downloaded by the OntoGene server, processed (in this case using the CTD entity vocabulary) and delivered to the user's browser. The user can then inspect all entity annotations and candidate interactions created by OntoGene.

The annotations are visible in two formats: either as highlighted text spans in the document panel or as a table in the annotation panel (the two panels are shown side by side). A customizable color-coding shows different entity types in the documents. Hovering the mouse over an annotated span will show the type and identifier values (IDs) of the annotation (IDs depend on specific database: CTD in this case). The concept panel shows all entity IDs that have been assigned to annotations in the document. The two panels are linked, so that when a user selects an item in the concept panel, the corresponding span(s) are highlighted in the document. We summarize the experience of the users in the rest of this section.

Since a given span could have multiple IDs (because of inherent ambiguity), and the same IDs could appear in several spans in the document, there is actually a many-to-many correspondence between items in the concept table and document spans. This aspect of the system was a bit confusing for some curators.

Items in the concept table can be easily sorted according to different criteria (*e.g.*, name, ID, frequency, type, etc). No problems were identified with this facility.

If the user enters an incorrect or non-existing PubMed ID, the system tries to download it from PubMed and appears to be processing it for a while, ending in a blank screen. The users correctly pointed out that an explanatory error message would be helpful.

All entity annotations can be edited: users can modify or remove existing terms and add new terms. Deletion of a new term is a trivial procedure. Modification of an existing term by type or ID is also relatively simple. Addition of a new term is simple as long as the new term does not overlap existing term annotations. In this case it is necessary to first delete the existing annotations in order to create the new one. This procedure was a source of some confusion. We intend to clarify it in future releases of ODIN.

Additionally ODIN provides a panel containing candidate interactions suggested by the system, and ranked according to an internal confidence score. While it is relatively easy for a user to inspect the interactions, and then confirm or reject them, the system still lacks a way to add completely new interactions. This is a planned extension in a forthcoming release of ODIN.

Currently it is possible to export selected entities or interactions as a plain text file, and the curators had no difficulty in performing this task. However it would of course be desirable to be able to export them also in other common formats (*e.g.*, Excel, BEL, etc.).

In general, all curators rated their experience with ODIN as either positive or very positive.

## **Conclusion and future work**

The experiment briefly described in this paper shows that ODIN is a user-friendly, easy-to-use web interface that can address some of the problems that curators are confronted with during their daily activities.

However, it also pointed out to some problems and shortcomings that are not due to any intrinsic limitation of the system but rather insufficient field-testing. Problems such as missing or unclear error messages can be easily solved by OntoGene programmers. Additional help menus for specific panels and tasks are already available and need only to be verified and switched on. The curators also suggested enhancements that will be considered going forward. One of these was expanding the abstract to include any MeSH terms not in the the text. Another was the capability to past in text blocks from any source.

In fact some of the feedback provided by curators during the experiments was used already to improve ODIN before the official termination of the task. A revised version was released early in September which took into account much of the feedback received up to that point. Aspects of the system that were improved include a more consistent color highlighting scheme, removal of some discrepancies in the manual, novel filters to allow focuses inspection of selected sentences. We believe that the experience was extremely positive for all parties involved and we thank the BioCreative organizers for offering us this chance to partner developers and users of biomedical text mining technologies.

## **References**

1. Socorro Gama-Castro, Fabio Rinaldi, Alejandra López-Fuentes, Yalbi Itzel Balderas-Martínez, Simon Clematide, Tilia Renate Ellendorff, Julio Collado-Vides. Assisted curation of growth conditions that affect gene expression in *E. coli* K-12. Proceedings of BioCreative 2013, Washington, October 2013.
2. Fabio Rinaldi, Thomas Kappeler, Kaarel Kaljurand, Gerold Schneider, Manfred Klenner, Simon Clematide, Michael Hess, Jean-Marc von Allmen, Pierre Parisot, Martin Romacker, Therese Vachon (2008). OntoGene in BioCreative II. *Genome Biology*, 2008, 9:S13, PMC2559984
3. Fabio Rinaldi, Simon Clematide, Yael Garten, Michelle Whirl-Carrillo, Li Gong, Joan M. Hebert, Katrin Sangkuhl, Caroline F. Thorn, Teri E. Klein, Russ B. Altman. *Using ODIN for a PharmGKB revalidation experiment. The Journal of Biological Databases and Curation*, Oxford Journals, 2012, bas021; doi:10.1093/database/bas021

4. Fabio Rinaldi and Simon Clematide and Simon Hafner and Gerold Schneider and Gintare Grigonyte and Martin Romacker and Therese Vachon. Using the OntoGene pipeline for the triage task of BioCreative 2012. *The Journal of Biological Databases and Curation*, Oxford Journals, 2013. doi:10.1093/database/bas053
5. Salgado H, Peralta-Gil M, Gama-Castro S, Santos-Zavaleta A, Muñiz-Rascado L, García-Sotelo JS, Weiss V, Solano-Lira H, Martínez-Flores I, Medina-Rivera A, Salgado-Osorio G, Alquicira-Hernández S, Alquicira-Hernández K, López-Fuentes A, Porrón-Sotelo L, Huerta AM, Bonavides-Martínez C, Balderas-Martínez YI, Pannier L, Olvera M, Labastida A, Jiménez-Jacinto V, Vega-Alvarado L, Del Moral-Chávez V, Hernández-Alvarez A, Morett E, Collado-Vides J. RegulonDB v8.0: omics data sets, evolutionary conservation, regulatory phrases, cross-validated gold standards and more. *Nucleic Acids Res.* 2013 Jan;41(Database issue):D203-13. doi: 10.1093/nar/gks1201. Epub 2012 Nov 29.
6. Davis AP, Murphy CG, Johnson R, Lay JM, Lennon-Hopkins K, Saraceni-Richards C, Sciaky D, King BL, Rosenstein MC, Wiegers TC, Mattingly CJ. The Comparative Toxicogenomics Database: update 2013. *Nucleic Acids Res.* 2013 Jan 1;41(D1):D1104-14.

# MarkerRIF: An Interactive Curation System for Biomarker

Hong-Jie Dai<sup>1\*</sup>, Chi-Yang Wu<sup>2</sup>, Wei-San Lin<sup>1</sup>, Richard Tzong-Han Tsai<sup>3</sup>, Wen-Lian Hsu<sup>2</sup>

<sup>1</sup>Graduate Institute of BioMedical Informatics, College of Medical Science and Technology, Taipei Medical University, Taipei, Taiwan, R.O.C., <sup>2</sup>Institute of Information Science, Academia Sinica, Taipei, Taiwan, R.O.C., <sup>3</sup>Department of Computer Science and Information Engineering, National Central University, Taoyuan, Taiwan, R.O.C.

\*Corresponding author: E-mail: [hjdai@tmu.edu.tw](mailto:hjdai@tmu.edu.tw)

## Abstract

Disease-related biomedical researches nowadays focus on uncovering biomarkers, which are genes and protein that can act as indicators of the biological state of the ongoing disorder. Observing the expression and behavior of biomarkers can greatly benefit clinical researches and decisions. To efficiently and precisely extract biomarker-related knowledge buried within biomedical texts, we developed a text mining-based curation system named MarkerRIF, which allows curators to retrieve biomarker-related narrations and store their annotations directly while browsing through PubMed.

## Motivation and Background

Entrez Gene is a repository for gene-specific knowledge of the National Center for Biotechnology Information (NCBI). In addition to general and genomic information, narrative evidences regarding the gene functions within publications can be found in the GeneRIF (Gene Reference Into Function) section. This section provides a platform that enables scientists to share and enrich gene-related functional annotations.

In view of GeneRIF, we developed a browser extension named BioMarker Reference Into Function (MarkerRIF), which allows users to view and edit gene-related functions described in the abstract instantly online. Replacement of the word “Gene” with “Marker” delivers the main purpose of our tool, which is to look for supporting evidence of disease biomarker candidates that were uncovered through previous text-mining processes.

MarkerRIF contains functions including gene name and disease term annotation, linking of the aforementioned terms to their corresponding database, and the extraction of MarkerRIF sentences. Furthermore, a user curation interface is available for curators to curate or modify the extracted RIF sentences. Once confirmed, users can also directly submit the function-describing sentence to our MarkerRIF database or the GeneRIF section of the Entrez Gene database to further elucidate the behavior of these genes. A collection of this knowledge from the literature

should provide additional help in the study of biomarkers and may supplement clinical decision making.

## **MarkerRIF Installation**

### **Google Chrome**

1. Download the file “mrif.crx” from <http://bws.iis.sinica.edu.tw/MarkerRIF/> or use the direct link <http://bws.iis.sinica.edu.tw/MarkerRIF/mrif.crx>.
2. Open your Google Chrome browser, and modify its settings from the upper right panel.
3. Go to the directory “Tools”, and the option “Extensions” under it.
4. Drag and drop “mrif.crx” onto the Extensions page, and a confirmation of adding this tool will appear in a few seconds.
5. When installation is complete, a new page delineating the changes of MarkerRIF will be shown for your reference.
6. Please make sure MarkerRIF is enabled under the Extensions page.

### **Mozilla Firefox**

1. Download the file “mrif.xpi” from <http://bws.iis.sinica.edu.tw/MarkerRIF/> or use the direct link <http://bws.iis.sinica.edu.tw/MarkerRIF/mrif.xpi>
2. Open your Firefox browser. Go to the directory “Tools”, and the option “Extensions” under it.
3. Drag and drop “mrif.xpi” onto the Extensions page, and a confirmation of adding this tool will appear in a few seconds.
4. After installing MarkerRIF, an X-shaped symbol will appear at the lower right corner of the browser.
5. Please make sure MarkerRIF is enabled under the Extensions page.

## **Usage Scenario**

### **Browsing with MarkerRIF**

1. Go to the Extensions of the Google Chrome/Firefox Browser.
2. Google Chrome: Click on “Options” under MarkerRIF, and you will be directed to a new page.  
Firefox: After installing MarkerRIF, a X-shaped symbol will appear at the lower right corner of the browser. Single click on the symbol, and a pop-up window will appear.
3. On this page/pop-up window, two steps are required to enable MarkerRIF. First, choose and load the gene list of interest of which you would like to observe its function in abstracts. An example biomarker gene list file can be downloaded from <http://bws.iis.sinica.edu.tw/MarkerRIF/default.glist>.
4. Following step 3, grant the access of your Google account to MarkerRIF.

- Google Chrome: On the same page where you loaded the gene list, click on “Grant Google Access” and you will be directed to a new page. Accept the request of MarkerRIF, and you will be redirected to the MarkerRIF extension page with your account name at the top.
- Firefox: On the same pop-up window where you loaded the gene list, click on “Grant Google Access” and you will be directed to a new page. Accept the request of MarkerRIF, and you will be redirected to the MarkerRIF pop-up window with your account name at the top.
- An alternative way of granting account access is also provided on <http://bws.iis.sinica.edu.tw/MarkerRIF/Account/Register>
- Note that after the grant, MarkerRIF can only access your full name and email information associated with your Google account, no further information will be accessed by MarkerRIF. Unauthorized users will not be able see the function-related sentences provided by MarkerRIF. Figure 1 shows the results after step 3 and 4.

**Figure 1.** Gene list loading and granting Google account access to MarkerRIF

**Welcome Johnny Wu**

Choose File default.glist Load

- **default.glist** (n/a) - 716 bytes, last modified: 5/30/2013
  - 57016 (aldo-keto reductase family 1, member B10 (aldose reductase))
  - 51280 (golgi membrane protein 1)
  - 8842 (prominin 1)
  - 1116 (chitinase 3-like 1 (cartilage glycoprotein-39))
  - 14734 (glypican 3)
  - 4072 (epithelial cell adhesion molecule)
  - 2719 (glypican 3)
  - 1499 (catenin (cadherin-associated protein), beta 1, 88kDa)
  - 3569 (interleukin 6 (interferon, beta 2))
  - 7015 (telomerase reverse transcriptase)
  - 3068 (hepatoma-derived growth factor (high-mobility group protein 1-like))
  - 1737 (dihydrolipoamide S-acetyltransferase)
  - 174 (alpha-fetoprotein)
  - 11576 (alpha fetoprotein)
  - 7422 (vascular endothelial growth factor A)
  - 213 (serum albumin)
  - 7157 (tumor protein p53)
  - 1261665 (telomerase)
  - 3481 (insulin-like growth factor 2 (somatomedin A))
  - 4684 (NCAM)

Grant Google Access Remove Grant

5. Visit PubMed, and search the website with predefined query terms (e.g. liver cancer). We have discussed and created a set of query terms with collaborated curators, and it can be found at <http://bws.iis.sinica.edu.tw/MarkerRIF>.

- When viewing the search results, please assign the format of the “Display Settings” as abstracts if you want to see the curation interface.
- For abstracts that were primarily uncategorized, MarkerRIF can automatically arrange them into four different sections: Objectives, Methods, Results and Conclusions. Automatically sectioned abstracts are preceded by a note to inform researchers that they were categorized by MarkerRIF. Biomedical named entities including gene names, and disease terms are highlighted in the displayed results with different colors, respectively. When the mouse cursor is moved over the recognized entities, a brief pop-up summary of each entity will be displayed as shown in Figure 2. In addition, these entities can be hyperlinked to their corresponding Entrez gene or MeSH pages.

**Figure 2.** PubMed search results marked up by MarkerRIF.

J Proteome Res. 2013 Mar 5. [Epub ahead of print]

**Quantitative Proteomic Analysis Identified Paraoxonase 1 in Hepatocellular Carcinoma.**

Huang C, Wang Y, Liu S, Ding G, Liu W, Zhou J, Kuang M, Ji Y, Kong L, et al. Liver Cancer Institute, Zhongshan Hospital and Shanghai Medical School of Life Science, Education, Shanghai, China.

**Abstract**

The following sections were categorized by MarkerRIF.

**OBJECTIVE** This study aimed to identify serum biomarkers for hepatocellular carcinoma (HCC).

**METHOD** MVI is a histological sign of micrometastasis in the liver. In this study, HCC patients with different vascular invasion statuses was examined. The extent of vascular invasion in the pooled sera was analyzed.

**RESULT** Western blot analyses in 90 HCC cases confirmed the correlation of the expression level of *paraoxonase 1* (*PON1*) with the extent of vascular invasion. ELISA assays demonstrated the diagnostic utility of the *PON1* level, with the area under curve values of 0.847 and 0.889 for the MVI and gross vascular invasion, respectively, relative to the patients without vascular invasion, in a cohort of 387 additional HCC cases. Immunohistochemistry revealed that *PON1* expression in tumor cells was inversely correlated with the extent of vascular invasion in 200 additional HCC cases. In conclusion, using a proteomic approach, we found that serum *PON1* was a novel diagnostic biomarker for MVI.

**CONCLUSION** The prognostic values of serum *PON1* and its possible therapeutic applications are worth further investigation.

PMID: 23442176 [PubMed - as supplied by publisher]

**Information from Entrez Gene**

**Official Symbol:** [PON1](#)  
**Official Full Name:** paraoxonase 1  
**RefSeq status:** REVIEWED  
**Organism:** *Homo sapiens*  
**Also known as:** ESA, MVCD5, PON

---

**Summary:**  
 The enzyme encoded by this gene is an arylesterase that mainly hydrolyzes paraoxon to produce p-nitrophenol. Paraoxon is an organophosphorus anticholinesterase compound that is produced in vivo by oxidation of the insecticide parathion. Polymorphisms in this gene are a risk factor in coronary artery disease. The gene is found in a cluster of three related paraoxonase genes at 7q21.3. [provided by RefSeq, Oct 2008]

- As shown in Figure 3, all probable RIF candidate sentences extracted by MarkerRIF will be listed on the bottom of each abstract, with ones that match the gene list arranged ahead of others.

### Curation with MarkerRIF

Figure 4 shows the curation interface of MarkerRIF. All sentences containing the gene name in an abstract are analyzed by our sentence determiner and scored by MarkerRIF, and suggested candidate sentences are listed at the bottom. The record of each sentence can be edited using the curation interface, which allows users to add a new RIF sentence, or to modify the content of an existing textual sentence and validate whether the sentence truly conveys RIF knowledge.

**Figure 3.** Candidate RIF sentences extracted by MarkerRIF.

GeneRIFs							Confirm All	Confirm Selected	+ Add new record
<input type="checkbox"/>	PMID	Gene ID	Name	Textual evidence	GeneRIF?	Confirmed			
<input type="checkbox"/>	23442176	5444	PON1	Immunohistochemistry revealed that PON1 expression in tumor cells was inversely correlated with the extent of vascular invasion in 200 additional HCC cases.	Yes	Confirmed!			
<input type="checkbox"/>	23442176	5444	PON1	In conclusion, using a proteomic approach, we found that serum PON1 was a novel diagnostic biomarker for MVI.	Yes	Confirmed!			
<input type="checkbox"/>	23442176	5444	PON1	The prognostic values of serum PON1 and its possible therapeutic applications are worth further investigation.	No	Confirmed!			
<input type="checkbox"/>	23442176	5444	Paraoxonase 1	Quantitative Proteomic Analysis Identified Paraoxonase 1 as a Novel Serum Biomarker for Microvascular Invasion in Hepatocellular Carcinoma.	Yes	Confirmed!			
<input type="checkbox"/>	23442176	5444	paraoxonase 1	Western blot analyses in 90 HCC cases confirmed the correlation of the expression level of paraoxonase 1 (PON1) with the extent of vascular invasion.	Yes	Confirmed!			
<input type="checkbox"/>	23442176	5444	PON1	ELISA assays demonstrated the diagnostic utility of the PON1 level, with the area under curve values of 0.847 and 0.889 for the MVI and gross vascular invasion, respectively, relative to the patients without vascular invasion, in a cohort of 387 additional HCC cases.	No	Confirmed!			

**Figure 4.** Curation interface of MarkerRIF

Furthermore, causes of false positive sentences are generally divided into four categories: non-RIF related, negation, entity recognition error, and others. Once the user confirms and saves the curation results, it will be submitted and stored on our server. Data provided from different users accounts are stored individually and can be used for additional comparison and analysis.

### **Proposed tasks and curators for the BioCreative user interactive task**

Users will be given a list of genes related to liver cancer, along with a total of three different sets of abstracts. For these abstracts, please extract the following information: PMID of the abstract, gene terms and its corresponding gene ID from Entrez Gene, evidence sentence containing RIF information, and relation assertion (descriptive of RIF or not). The task will be run both manually and using MarkerRIF.

- Manual task: Curators will be assigned a set of PubMed abstracts for further processing, and should submit their annotations manually at <http://bws.iis.sinica.edu.tw/MarkerRIF/Annotation/Create> as shown in Figure 5.
- Using MarkerRIF: In contrast to the manual task, curators will extract the information of interest from the two other set of abstracts with the assistance of MarkerRIF, The curated results will be stored and accessible through the MarkerRIF website upon logging in (<http://bws.iis.sinica.edu.tw/MarkerRIF/Account/LogOn>). Curators can then compare and analyze the differences between the two approaches, and offer suggestions for further improvement.

**Figure 5.** The manual curation interface.

The screenshot shows a web interface for creating an annotation. At the top, there are three navigation tabs: 'Home', 'Annotations', and 'About'. The main content area is titled 'Create' and contains a form with the following fields:

- Annotation** section:
  - PMID: 23442176
  - Gene ID: 5444
  - Gene Name: PON1
- Textual evidence**: A text area containing the sentence: "Immunohistochemistry revealed that PON1 expression in tumor cells was inversely correlated with the extent of vascular invasion in 200 additional HCC cases." The words 'Immunohistochemistry' and 'HCC' are underlined in red.
- Comment**: A dropdown menu currently set to 'None'.
- IsGeneRIF**: A checked checkbox.
- A 'Create' button at the bottom of the form.

### Details of the protocol

Input: Assigned set of specific disease-related abstracts.

Output: Output of the extracted information should be presented accordingly to the following tab-delimited format:

PMID | Gene ID | Gene name | Evidence sentence | Relation assertion

### Curation dataset selection

To perform the proposed curation task, a curation dataset along with a list containing genes of interest is provided. The gene list contains probable liver cancer biomarkers collected from several review papers (12, 13, 14). The curation dataset consists of 190 abstracts retrieved from PubMed using two different query terms. One is the predefined query listed on our website

((blood[Title/Abstract] OR serum[Title/Abstract] OR urine[Title/Abstract]) AND clinical[Title/Abstract]) OR diagnosis[Title/Abstract]) AND liver cancer[Title/Abstract],

and the other being

(carcinoma, hepatocellular[MeSH Terms]) AND biomarker.

The former query term is defined by domain experts in search of their information of interest within abstracts, and the latter is a more straightforward query used to look for liver cancer biomarkers. The 190 abstracts are divided into three sets in correspondence with the three curators participating in the full biocuration, with each containing 63, 63, 64 abstracts, respectively. Each curator will perform complete manual curation on one set, and MarkerRIF-assisted curation on the other two sets. Due to the limit of time, we have asked the curators to only curate the first 30 abstracts of each set, with a total of 90 curated abstracts for each curator.

## **Technical Details**

### **Text-mining web server and system performance**

The text-mining server comprises three REpresentational State Transfer (REST) architectural web services.

### ***Section Categorizer***

The section categorizer demarcates abstracts into different paragraphs regarding their content. For a given abstract, if PubMed or the pre-sectioned check uncovers that the abstract does not contain obvious section tags, such as ‘Objective’ and ‘Conclusion’, a machine learning-based categorizer (1) is employed to dissect the given abstract.

### ***Named Entity Tagger***

The service include two named entity taggers. The first is a machine learning-based gene mention tagger (2), which labels gene names in abstracts. Following entity recognition, an entity normalization module normalizes the found gene names to their corresponding Entrez Gene database identifiers using a multi-stage approach (3). Our gene mention tagger achieved an F-score of 86.24% on the BioCreAtIvE II corpus (4, 5). The performance of our normalization system was evaluated on our instance-based gene mention linking corpus (6), and achieved F-scores of 0.856 and 0.71 for human genes on the article-wide and the instance-based levels, respectively. For cross species evaluation, it achieved the highest area under the precision/recall curve score (0.58) on the BioCreative II.5 interactor normalization dataset (7) and the threshold average precision score of 0.413, which used the median of the confidence scores among all 20<sup>th</sup> instances as the threshold; The system ranked second in the BioCreative III gene normalization dataset(8).

The second tagger is a dictionary-based disease name tagger based on the maximum matching algorithm. We compiled a dictionary of about 40,000 disease terms with corresponding unique identifiers from the MeSH database. It achieved a satisfactory F-score of 83.4% on the Jimeno *et al.*'s corpus (9).

### ***Sentence Determiner***

The sentence determiner provides evidence sentences for genes of interest at the bottom of the abstract. A list of RIF related terms, such as “downregulate” and “induce”, is organized, and sentences containing both the gene name and RIF related terms are extracted and ranked by a machine learning model. Several works have proposed effective features in GeneRIF indexing, such as (10, 11). This work focuses on biomarker-related narratives. We hope to evaluate the effective of the employed features in the specific task by participating the interactive track. In respect of valuable feedbacks, we constructed a user friendly interface for users to curate these sentences and express their thoughts.

### **MarkerRIF client interface and database**

The client interface of MarkerRIF is mainly written in JavaScript with Google Chrome application programming interface and the add-on software development kit of Mozilla Firefox. The OAuth 2.0 authorization framework<sup>1</sup> is employed to obtain curators' profile information to reduce the effort of user registration. The MarkerRIF database is set up on a Windows server with ASP.NET, MongoDB and SQL server.

### **Preliminary Curation Results**

A total of three curators were involved in the full participation of the biocuration track. Three sets of liver cancer-related abstracts, each containing 30 abstracts, were assigned to the curators for manual or MarkerRIF-assisted curation, respectively. Judging by current annotations, the notion and function of MarkerRIF were well understood by the participating curators. For instance, the sentence “Additionally, the expression characteristics of annexin A2 during hepatocarcinogenesis were detected in p21-HBx gene knockin transgenic mice model.” were deemed as “Non-RIF-related” regarding annexin A2 by two of the curators. The sentence “Between January 2003 and December 2005, we enrolled 115 treatment-naive patients who received **TACE** as an initial treatment modality.” was marked as “Entity recognition error”, since TACE is a treatment rather than a gene in this case. As for the sentence “Expression of BRM mRNA, but not BRG1 mRNA, was significantly reduced in primary HCC tumours, compared to non-tumour tissue counterparts.”, it was considered as “Negation” regarding BRG1, since its expression remains unaffected in primary HCC tumours. After curators complete the task, we will analyze the inter-annotator agreement to validate the consistency of these annotations and then report the overall evaluation results of our biomarker-related sentence extraction.

---

<sup>1</sup> <http://tools.ietf.org/html/rfc6749>

## Acknowledgements

This work was supported by the National Science Council of Taiwan under the grant number NSC102-2319-B-010-002, NSC-102-2218-E-038-001, and the Taipei Medical University under the grant number TMU101-AE1-B55.

## References

1. R. T. K. Lin, H.-J. Dai, Y.-Y. Bow, J. L.-T. Chiu, and R. T.-H. Tsai, "Using conditional random fields for result identification in biomedical abstracts " *Integrated Computer-Aided Engineering*, vol. 16, pp. 339-352, Dec. 2009.
2. R. T.-H. Tsai, C.-L. Sung, H.-J. Dai, H.-C. Hung, T.-Y. Sung, and W.-L. Hsu, "NERBio: using selected word conjunctions, term normalization, and global patterns to improve biomedical named entity recognition," *BMC Bioinformatics*, vol. 7, p. S11, Dec. 2006.
3. H.-J. Dai, P.-T. Lai, and R. T.-H. Tsai, "Multistage Gene Normalization and SVM-Based Ranking for Protein Interactor Extraction in Full-Text Articles," *IEEE/ACM TRANSACTIONS ON COMPUTATIONAL BIOLOGY AND BIOINFORMATICS*, vol. 7, pp. 412-420, 2010.
4. L. Smith, L. K. Tanabe, R. J. n. Ando, C.-J. Kuo, I.-F. Chung, C.-N. Hsu, Y.-S. Lin, R. Klinger, C. M. Friedrich, K. Ganchev, M. Torii, H. Liu, B. Haddow, C. A. Struble, R. J. Povinelli, A. Vlachos, W. A. B. Jr, L. Hunter, B. Carpenter, R. T.-H. Tsai, H.-J. Dai, F. Liu, Y. Chen, C. Sun, S. Katrenko, P. Adriaans, C. Blaschke, R. Torres, M. Neves, P. Nakov, A. Divoli, M. Maña-López, J. Mata, and W. J. Wilbur, "Overview of BioCreative II gene mention recognition," *Genome Biology*, vol. 9, p. S2, 2008.
5. A. A. Morgan, Z. Lu, X. Wang, A. M. Cohen, J. Fluck, P. Ruch, A. Divoli, K. Fundel, R. Leaman, J. Hakenberg, C. Sun, H.-h. Liu, R. Torres, M. Krauthammer, W. W. Lau, H. Liu, C.-N. Hsu, M. Schuemie, K. B. Cohen, and L. Hirschman, "Overview of BioCreative II gene normalization," *Genome Biology*, vol. 9, p. S3, 2008.
6. H.-J. Dai, Y.-C. Chang, R. T.-H. Tsai, and W.-L. Hsu, "Integration of gene normalization stages and coreference resolution using a Markov logic network," *Bioinformatics*, vol. 27, pp. 2586-2594, 2011.
7. F. Leitner, S. A. Mardis, M. Krallinger, G. Cesareni, L. A. Hirschman, and A. Valencia, "An Overview of BioCreative II.5," *IEEE/ACM TRANSACTIONS ON COMPUTATIONAL BIOLOGY AND BIOINFORMATICS*, vol. 7, pp. 385-399, 2010.
8. Z. Lu, H.-Y. Kao, C.-H. Wei, M. Huang, J. Liu, C.-J. K. C.-N. Hsu, R. T.-H. Tsai, H.-J. Dai, N. Okazaki, H.-C. Cho, M. Gerner, I. Solt, S. Agarwal, F. Liu, D. Vishnyakova, P. Ruch, M. Romacker, F. Rinaldi, S. Bhattacharya, P. Srinivasan, H. Liu, M. Torii, S. Matos, D. Campos, K. Verspoor, K. M. Livingston, and W. J. Wilbur, "The gene normalization task in BioCreative III," *BMC Bioinformatics*, vol. 12, p. S2, 2011.
9. A. Jimeno, E. Jimenez-Ruiz, V. Lee, S. Gaudan, R. Berlanga, and D. Rebholz-Schuhmann, "Assessment of disease named entity recognition on a corpus of annotated sentences," *BMC Bioinformatics*, vol. 9, p. S3, 2008.
10. A. Jimeno-Yepes, J. Sticco, J. Mork, and A. Aronson, "GeneRIF indexing: sentence selection based on machine learning," *BMC Bioinformatics*, vol. 14, p. 171, 2013.

11. Z. Lu, K. B. Cohen, and L. Hunter, "Finding GeneRIFs via gene ontology annotations," *Pac Symp Biocomput*, pp. 52-63, 2006.
12. T. Behne and M. Sitki Copur, "Biomarkers for Hepatocellular Carcinoma," *International Journal of Hepatology*, vol. 2012, Article ID 859076, 7 pages, 2012.
13. A. Singhal, M. Jayaraman, D. N. Dhanasekaran, and V. Kohli, "Molecular and serum markers in hepatocellular carcinoma: Predictive tools for prognosis and recurrence," *Critical Reviews in Oncology/Hematology*, vol. 82, pp. 116–140, May 2012
14. R. Masuzakia, S. J. Karpa, M. Omata, "New Serum Markers of Hepatocellular Carcinoma," *Seminars in Oncology*, vol. 39, pp. 434-439, Aug. 2012

# Supporting Document Triage with the SciKnowMine System in the Mouse Genome Informatics (MGI) Curation Process

Gully APC Burns<sup>1</sup>, Marcelo Tallis<sup>1</sup>, Hiroaki Onda<sup>2</sup>, Kevin Cohen<sup>3</sup>, James Kadin<sup>2</sup>, Judith Blake<sup>2</sup>

<sup>1</sup> USC Information Sciences Institute, Marina del Rey, CA 90292

<sup>2</sup> Jackson Laboratory, Bar Harbor, ME, Jackson Laboratory, Bar Harbor, ME 04609

<sup>3</sup> University of Colorado School of Medicine, Aurora, CO 80045

## Abstract

We describe ‘SciKnowMine’: a software-driven platform for delivering document triage functionality in an extensible web-based biocuration system. The system was designed principally to provide an extensible platform that could be instantiated with any machine learning model document triage as needed. At this stage we emphasized the design of the underlying data structures supporting the triage task and the systems-administration functionality needed to build a deployable system. We instantiated this system at the Mouse Genome Informatics group at the Jackson Laboratory through August / September of 2013 and report here the outcomes of this deployment.

## Relevance and Impact

### Background

MGI has been one of the foremost biomedical informatics groups that actively support the development of text mining systems as part of their biocuration workflow. They provided data for shared task challenges for the TREC community concerned specifically with document triage of biomedical text in 2004-5 (1, 2) and have invested their support in the community actively since then. We regard the fact that MGI does not have any functional text mining tools for use in their everyday curation work as a serious failure for the field as a whole. We therefore focus our efforts on: (a) developing a well-designed base-level application that reuses simple, well-tested classification technology; (c) deploying this system to MGI so that it could be run from within MGI with minimal support from our team. Within the preliminary work described here, we present the SciKnowMine Triage Web Application (SKM-TWA) as an open-source system for consideration by the community (3).

At the time of writing, (June 7th 2013), we have just deployed a working first prototype of the system as a robust virtual appliance for use by MGI curators). We present this write-up as a contribution to the BioCreative 4 IAT challenge because the key issues that determine success in our work have been primarily infrastructural, logistical, and engineering-driven rather than a detailed consideration of the algorithmic approach being used. These are not issues being

considered within the IAT evaluation. We also argue that the issue of performing extrinsic evaluation within the context of an actual curation team performing a curation task remains an extremely poorly defined problem. Our current implementation falls short of required criteria in the statement of the challenge, but we submit our work for consideration by the community in the hope of fostering debate and providing a concrete implementation of a possible solution that perhaps has emphasized criteria that were not explicitly highlighted within the challenge specification.

### **Applying Machine Learning (ML) to Document Triage**

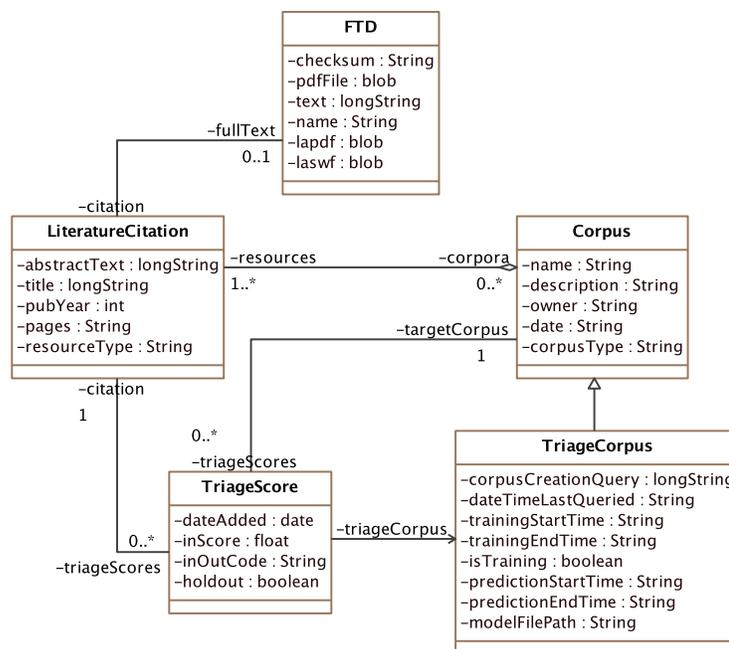
Document triage is a preliminary task in MGI's curation pipeline. It is a prime candidate for automation within the overall workflow of the system. The impact of making mistakes at this stage is significant since false negatives entail that relevant papers are missed. These missed papers would never be picked up by other checks or failsafes in the MGI curation pipeline and would only be corrected if an external party reported the omission to the curation team. Thus systems with high-recall are a priority.

We view the underlying computational problem as one of Document Classification (4): given a full-text scientific article in the form of a PDF file, can we computationally determine whether it should be judged 'in' or 'out' of our curation set? We are not here considering the logistical issues of obtaining the articles in question (including questions of licensing and access to articles for bulk downloads for evaluation). MGI licenses all journals that they typically curate from and negotiate bulk download privileges from publishers.

Since MGI is already a large scale database, We have a large quantity of preexisting material which has already been manually classified (183,371 separate article citations are currently listed in MGI) providing a corpus of training material that makes using supervised learning a good candidate approach. Support Vector Machines are a well-established technology for document classification (5) and we use LibSVM (6) implementation supported within the ClearTk software package (7, 8).

### **User Interactivity**

A fundamental aspect of our software's usability is how we frame the logic of the triage problem itself. Although our model is not directly visible to users, it strongly guides users' experience and so we present it here.



**Figure 1.** UML Model for the triage system

### A Design Pattern for Document Triage

Figure 1 shows a UML data model for SKM-TWA as a conceptual abstraction of triage data. In this design, a **Corpus** may be defined as any collection of **LiteratureCitation** objects (some of which are supported by **FTD** or ‘Full Text Documents’). The crucial element in this design is a **TriageScore** which is a ternary relationship between (a) **LiteratureCitation** objects, (b) **TriageCorpus** objects that each denote the collection of citations being classified and (c) target **Corpus** objects which each denote the set that the citations are being classified into. In this particular model, each **TriageScore** may be assigned an ‘inOutCode’ (as ‘in’, ‘out’ or ‘unclassified’) and a numerical ‘inScore’ float value ranging from [0,1]. This score is calculated by the execution of an SVM over features that are transiently generated from the text of a citation’s **FTD** object and are not explicitly stored in this model. The construct as shown also provides a framework for the easy addition of training data to a preexisting data set simply by assigning inOutCode values and then retraining the ML model.

### Use of PDFs

Working with the full text of papers in biocuration tasks has expectedly been shown to improve biocuration performance (9), but common issues (such as distributed storage, licensing restrictions and PDF formatting of full text articles) cause difficulties for developers of curation systems. We focus on working with text extracted from PDF files using our layout-aware PDF text extraction software (10) based on the idea of defining ‘active folders’ within the system’s server so that curators simply save PDF’s of interest into that folder for the system to begin

processing them. In one sense, processing PDFs directly provides a processing bottleneck that simplifies the system's design since all full text articles are published in that form.

## Design of the user interface

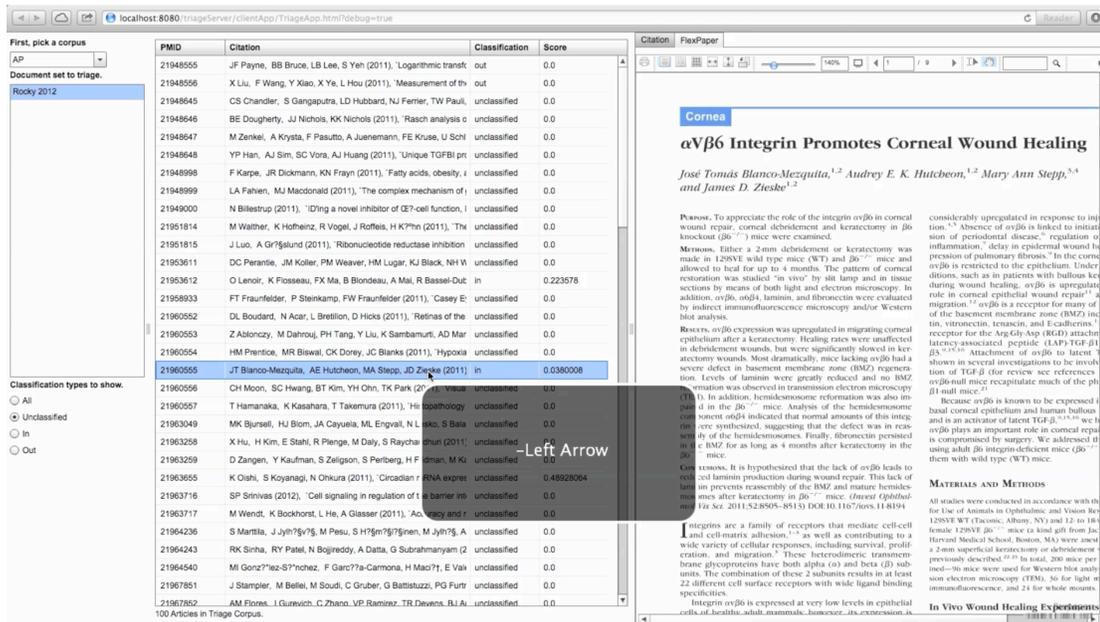


Figure 2. Screenshot of the triage system

Figure 2 shows a screen-capture of the user interface taken during a movie of the system's use in a demonstration<sup>1</sup>. Note that the grey box with the text reading '-Left Arrow' is a feature of the tool used to perform the video capture to show that the user has just pressed the left cursor key.

The workflow of usage within this demonstration is as follows:

1. Select the target **Corpus** in the top left hand combo box ('AP' stands for 'Allele Phenotype').
2. Select the document set (**TriageCorpus**) in the list beneath it. The system will then load all citations in the selected document set and will provide `inOutCode` and `inScore` values if present.
3. Upon selection of a citation in the list, the system will load the individual citation or the PDF file in the right hand panel. PDFs are viewed in a 'FlexPaper' component (which is actually a rendition of a \*.swf file that generated and stored in the database when articles are uploaded).
4. A curator may filter the list of scored citations in the middle panel based on their `inOutCode` values by using the radio buttons on the left.
5. A curator may scan through citations using the 'up' and 'down' cursor keys and may alter the `inOutCode` value of each citation in the list by pressing the left or right cursor (left

<sup>1</sup> <http://www.youtube.com/watch?v=NIUKYF-x5Gk>

for ‘in’ and right for ‘out’). This allows the curator to work through the content of a given `TriageCorpus` *without using the mouse*.

This provides a relatively simple, straightforward interaction mechanism for the curator to look through a corpus, examine the results of text mining processing (which in our case is only limited to `inScore` values and the PDF itself. We expect to show the most prominent features used by the ML analysis to the biocurator in the right hand panel, but have not yet implemented this.

### Preparing data in the system

We currently provide several data processing functions only as command line calls rather than being accessible through the user interface. This is to restrict access to the database only for an administrator (not a curator) working on the server. These commands are performed offline and are as follows:

1. `buildTriageDatabase`  
This generates a new MySQL-formatted triage database.
2. `editArticleCorpus`  
This creates a named target corpus within the database.
3. `editTriageCorpus`  
This creates a named target corpus within the database.
4. `addPmidEncodedPdfsToCorpus`  
This loads all PDF files found in the PDF directory into the specified triage corpus.
5. `buildTriageCorpusFromPmidList`  
This loads a formatted text file into `inOutCode` values for the given corpora. In this plain-text file each line consists of the pubmed-id of each file a ‘TAB’ and either a ‘+’ symbol for ‘in’, a ‘-’ symbol for ‘out’ or a ‘?’ symbol for ‘unclassified’ values.
6. `trriageDocumentsClassifier -train`  
This uses the data in the database to train a SVM model based on a preset hand chosen set of parameters we currently hard-code into the system.
7. `trriageDocumentsClassifier -predict`  
This uses the model file generated in the last step to generate `inScore` values for each citation in the specified **TriageCorpus**.

### Separation between ML experiments and system execution

An overlooked class of user of NLP-enabled biocuration systems is the class of NLP scientists attempting to run experiments on text within specific corpora in order to engineer classification features. We specifically provide command-line functions to these users (not shown) that query a named **TriageCorpus** to dump the text of each file to a separate line in a text file that is formatted in a way typically used by NLP experts for machine learning work. In this process, our system also randomly selects a proportion of papers to be used as a ‘held-out’ testing set and

dumps the text of those documents to another file. These files are placed into a separate subdirectory and may be processed using provided ML NLP packages such as ClearTk (7, 8).

### **The absence of annotations and other key differences**

The specification for the IAT competition specifies a need for ‘annotations’ on the original document. Given that the only feedback we are currently generating for curators to use to support their triage decision is a single score generated by bigram features, we assert that annotations over the text of the document would likely need to be based on the most influential features used by the ML process to generate a score. This remains an unfinished next-step of our current implementation.

### **System Performance**

At this stage of systems development (initial deployment of a prototype), we have not yet gathered extensive feedback from curators and systems administrators but share first impressions based on processing a test-corpus of 1000 documents. Loading documents into the database takes 6-10 seconds per PDF file, meaning that a document set of 1000 papers took roughly 3 hours to load (this also includes text-extraction and generation of \*.swf files). The execution of the training phase to generate a model from this corpus was relatively quick as was the application of that model to new data (10-15 minutes for each task over 1000 papers). We are currently at the stage of having deployed our preliminary demonstration system to the Jackson Laboratory curation team who are engaged with the process of running the system as prescribed as part of the IAT challenge. The deployment itself has been challenging and we report only partial success at running SciKnowMine within a real biocuration environment. Future work includes the addition of annotation functionality to PDFs and ways of instrumenting the software to evaluate accuracy and utility for the triage task going forward.

### **Funding**

This work was supported by the National Science Foundation [grant number #0849977 to GAPCB] and the National Institutes of Health [grant numbers 1R01MH079068-01A2 to GAPCB, RO1-GM083871 to GAPCB]

### **References**

1. A. M. Cohen and W. R. Hersh. The trec 2004 genomics track categorization task: classifying full text biomedical documents. *J Biomed Discov Collab*, 1:4, 2006. 1747-5333 (Electronic) Journal Article.
2. W. Hersh, A. Cohen, J. Yang, R. T. Bhupatiraju, P. Roberts, and M. Hearst. Trec 2005 genomics track overview. In *Text REtrieval Conference (TREC) 2005*, Gaithersburg, Maryland, 2005.
3. Cartic Ramakrishnan, William A. Baumgartner Jr., Judith Blake, Gully APC Burns, K. Bretonnel Cohen, Harold Drabkin, Janan Eppig, Eduard Hovy, Chun-Nan Hsu, Lawrence E. Hunter, Tommy Ingulfsen, Kevin Livingston, Hiroaki ‘Rocky’ Onda, Sandeep Pokkunuri, Ellen Riloff, Christophe Roeder, and Karin

- Verspoor. Building the scientific knowledge mine (sciknowmine1): a community-driven framework for text mining tools in direct service to biocuration. In Language Resources and Evaluation (LREC), Malta, 2010.
4. Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47, March 2002.
  5. N Christianini and J Shawe-Taylor. Support Vector Machines and other kernel based learning methods. Cambridge University Press, Cambridge, 2000.
  6. Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. Technical report, Department of Computer Science, National Taiwan University, 2007.
  7. Philip V. Ogren, Philipp G. Wetzler, and Steven Bethard. ClearTK: A UIMA toolkit for statistical natural language processing. In Towards Enhanced Interoperability for Large HLT Systems: UIMA for NLP workshop at Language Resources and Evaluation Conference (LREC), 5 2008.
  8. Philip V. Ogren, Philipp G. Wetzler, and Steven J. Bethard. ClearTK: a framework for statistical natural language processing. In Unstructured Information Management Architecture Workshop at the Conference of the German Society for Computational Linguistics and Language Technology, 9 2009.
  9. D. P. Corney, B. F. Buxton, W. B. Langdon, and D. T. Jones. Biorat: extracting biological information from full-length papers. *Bioinformatics*, 20(17):3206–13, 2004. 1367-4803 Evaluation Studies Journal Article.
  10. Cartic Ramakrishnan, Abhishek Patnia, Eduard Hovy, and Gully Burns. Layout aware text extraction from full-text PDF of scientific articles. *Source Code for Biology and Medicine*, 7(1):7, 2012.

# BioQRator: a web-based interactive biomedical literature curating system

Dongseop Kwon<sup>1</sup>, Sun Kim<sup>2,\*</sup>, Soo-Yong Shin<sup>3</sup>, and W. John Wilbur<sup>2</sup>

<sup>1</sup>Department of Computer Engineering, Myoungji University, South Korea

<sup>2</sup>National Center for Biotechnology Information, National Institutes of Health, USA

<sup>3</sup>Department of Biomedical Informatics, Asan Medical Center, South Korea

\*Corresponding author: Tel: 301 496 2484, E-mail: sun.kim@nih.gov

## Introduction

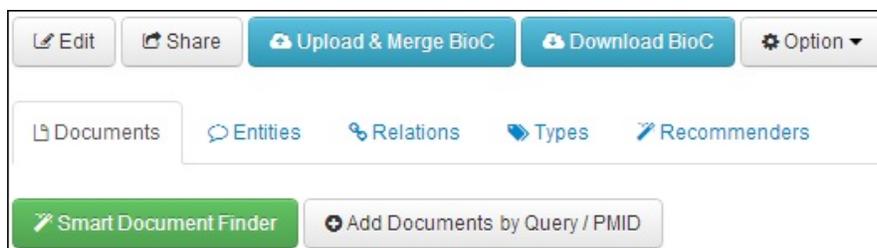
BioQRator (<http://www.bioqrator.org>) is a web-based annotation tool for biomedical literature. This tool was designed to support any task annotating entities and relationships. It is also one of the first web tools which support the BioC format (1) for annotation. For input, any documents in the BioC format and PubMed<sup>®</sup> abstracts can be used. For output, annotated documents can be saved in a BioC format file as well. Our goal in the BioCreative IV IAT task focuses on the following two topics.

- 1) Develop a general-purpose annotation tool for entities and relationships. This tool is essentially a web interface which can be fully customized for a given task. To assist an annotation task, text mining resources can be utilized through the BioC format.
- 2) Apply and evaluate PIE *the search* (2) for a protein-protein interaction (PPI) annotation task. PIE *the search* is a web interface for searching PubMed literature for protein interaction information and the main method is based on a winning approach in BioCreative III (3). In BioCreative IV IAT, the practical usability of PIE *the search* will be studied.

Here, we show basic functions of BioQRator and the performance of PIE *the search*. In addition, we propose a PPI annotation task for the BioCreative IV IAT task.

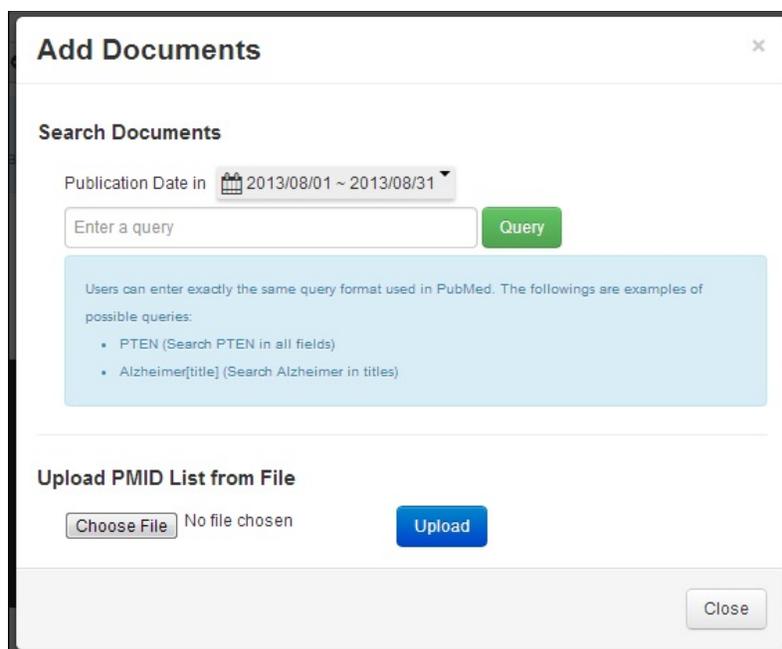
## System Description

BioQRator was designed as an easy-to-use tool to annotate any entities and relationships in text. In particular, most annotations can be done by a series of single mouse clicks (or drags) with simple typing. Since BioQRator was implemented using HTML5/CSS to support multiple browsers. It is compatible with the latest version of browsers such as Chrome, Safari and Firefox. Here is the scenario of how to use BioQRator.

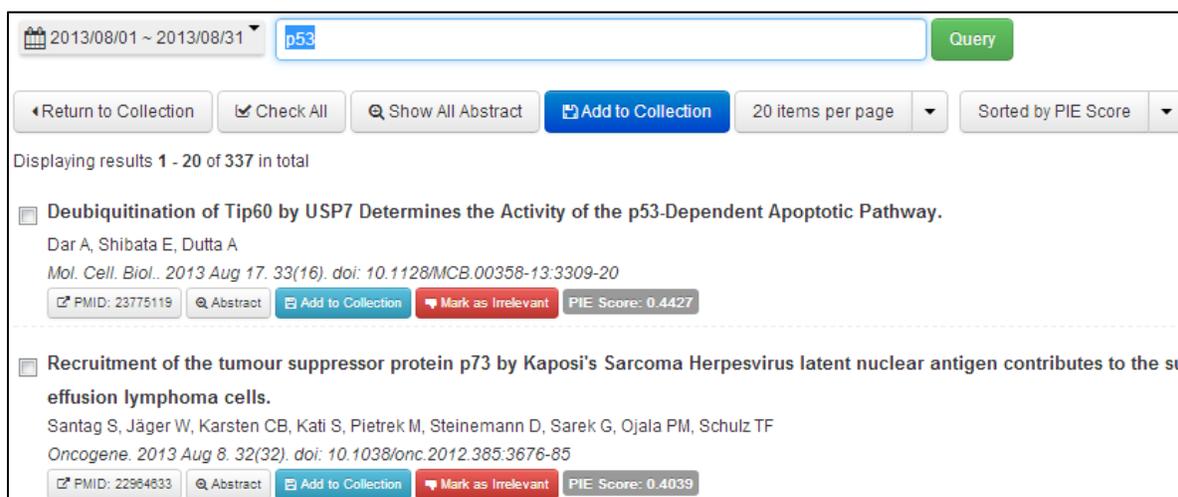


**Figure 1.** The main window of an empty collection.

- 1) Sign in (or sign up if there is no account)
- 2) Create a collection: A user can create a collection by several different methods.
  - A. From a web browser: A collection name is required. Source, date and key information can be optionally entered.
  - B. From a BioC format file: All necessary information including pre-annotated documents is automatically loaded using the uploaded BioC file.
- 3) Create entity and relation types: Given a collection, the next task is to associate with the collection those entity and relation types which will be used to annotate the collection. A user can create the entity and/or relation types by going to the “Types” tab (Figure 1) or during document annotation. The “Recommendors” tab (Figure 1) is used for setting up external resources to find relevant information for an entity name. However, BioQRator supports Entrez Gene and UniProt Recommendors in default.
- 4) Add documents
  - A. Unless documents are loaded from a BioC file, a user should add documents into an empty collection. Currently, we provide two options: “Search documents with a PubMed query” and “Upload a PMID list from a file” (Figure 2). Both options retrieve documents from PubMed, however retrieval results are sorted by PIE score in default. A higher PIE score means there is more possibility that the document may have PPI information.
  - B. Smart document finder (Figure 1): This is a convenient tool for periodically adding documents with a fixed query. A user will be able to set automatic document search weekly, monthly, quarterly, or even yearly.
  - C. After searching PubMed or PIE *the search*, a user can manually select and add any documents of interest by clicking “Add to Collection” (Figure 3). The “Abstract” button below each document title can be used for a quick look at abstracts in the same window. The “PMID: 23775119” button is used for reading abstracts/full text through the PubMed service. “Mark as Irrelevant” is a special feature in the BioQRator search. If a document is marked as irrelevant, it will not appear in future retrieval results in the same collection.
  - D. Adding documents is flexible. New documents or BioC files can be added to an existing collection any time.



**Figure 2.** Adding documents to a collection.



**Figure 3.** PubMed search results.

5) Annotate documents

- A. After adding a set of documents to form a collection, a user can start annotating entities and relations. For uploaded BioC files, pre-annotated entities and relations will be automatically shown in the annotation window.
- B. For annotating an entity, a user can do a single click or drag the mouse to select the whole entity name. Once a mouse click or drag is done, a pop-up window will appear and a user can fill in necessary information. For normalizing gene/protein

names, Entrez Gene and UniProt searches are provided in default. Entrez Gene or UniProt IDs can be easily assigned through this search process. Note that “Annotation ID” in this window is different from Entrez Gene or UniProt IDs. The annotation ID identifies the annotation uniquely and does not represent an ID in a database such as Entrez Gene or UniProt IDs. Normally, a user does not need to assign annotation IDs because BioQRator automatically assign the IDs unless specified.

- C. For PubMed abstracts, pre-annotated PPI entities will be available. A user can use this information by clicking “Open PIE *the search* Annotations” (Figure 4).

The screenshot displays the BioQRator interface. On the left, there is a text area with a 'title' tab and an 'abstract' tab. The title is 'Evaluation of Nod-like receptor (NLR) effector domain interactions.' The abstract text is partially visible, mentioning NLR signaling and protein interactions. On the right, there are tabs for 'Entities', 'Relations', and 'Types'. Below these tabs is a table with the following data:

ID	Type	Location	Text
A1	Protein	931:5	NLRP1
A2	Protein	948:6	NLRP12
A3	Protein	734:5 938:5	NLRP3
A4	Protein	696:4	NOD1
A5	Protein	705:4 830:4 885:4 1051:4	NOD2
A6	Protein	578:5 715:5 1005:5 1086:5	RIPK2

At the bottom of the interface, there is a button labeled 'Open PIE the search Annotations'.

**Figure 4.** Annotating entities and relations.

- 6) Download a collection: Annotated documents in a collection can be saved as a BioC format file. BioC was developed to easily share text documents and annotations among different tools. Since BioCreative IV took the BioC initiative as one of its main tasks, we decided to fully support BioC as the standard input and output file format.
- 7) Share a collection: A collection can be shared with other users. This function is enabled if other users are added through the “Share” button in a collection (Figure 1).

## Performance of PIE *the search*

To support PPI annotations, article ranking and entity information from PIE *the search* was migrated to BioQRator. In previous work (2), we evaluated article ranking performance using the BioCreative III ACT (BC3) dataset (4). For F1, MCC and AUC iP/R measures, PIE *the search* showed 0.6258, 0.5610 and 0.6834 respectively. However, the medians of BC3 participant results were 0.5353 F1, 0.4563 MCC and 0.5367 AUC iP/R. Table 1 shows the precisions of PIE *the search* at rank  $N$  for the BC3 test set. Since PubMed abstracts can be sorted based on PPI scores in BioQRator, the performance at top-ranked documents is more important than overall classification performance in this regard. Hence, the table shows the usefulness of PIE *the search* as a PPI informative article search tool.

**Table 1.** Ranking performance of PIE the search.

Top N	Precision
10	1.0000
50	0.9600
100	0.9400
200	0.9150
300	0.8467
400	0.8125
500	0.7680

For identifying gene/protein names, the Priority Model (5) is utilized in PIE *the search*. Since not all entities are important in PPI annotations, we only mark predicted gene/protein names which are used to identify PPI informative articles. In (5), the Priority Model showed 0.9200, 0.9690 and 0.9440 for precision, recall and F1 scores respectively on the experiments using SemCat (6).

## Proposed Tasks for BioCreative IV Track 5 (IAT)

For BioCreative IV IAT, our focus is on two goals: the usability of BioQRator as a general-purpose annotation tool and the effectiveness of PIE *the search* as a supporting tool for PPI annotations. To achieve these goals, the proposed tasks are as follows:

- 1) Search PubMed abstracts and sort the results based on relevance to PPI information: Ranking performance can be used as an evaluation measure. Search results obtained from BioQRator will be compared with those from PubMed.
- 2) Annotate PPI-relevant interactions and normalize protein names: Manual annotations are a time-consuming task. Reducing annotation time by using BioQRator is a main interest in the proposed task.
- 3) BioC compatibility: Supporting BioC as standard input and output format does not solve

all the interoperability issues. Synchronizing locations of entities and character codes (e.g., UTF-8 and ASCII) among different tools is a crucial problem. We plan to address these issues by communicating with other BioC developers.

## Acknowledgements

DK and SYS were supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2012R1A1A2044389 and 2011-0022437; 2012R1A1A2002804), respectively. SK and WJW were supported by Intramural Research Program of the NIH, National Library of Medicine.

## References

1. Comeau,D.C., Dogan,R.I., Ciccarese,P. *et al.* (2013) BioC: a minimalist approach to interoperability for biomedical text processing. *Database*, **2013**, bat064.
2. Kim,S., Kwon,D., Shin,S.-Y. *et al.* (2012) PIE *the search*: searching PubMed literature for protein interaction information. *Bioinformatics*, **28**(4), 597-598.
3. Kim,S. and Wilbur,W.J. (2011) Classifying protein-protein interaction articles using word and syntactic features. *BMC Bioinformatics*, **12**(Suppl 8), S9.
4. Krallinger,M., Vazquez,M., Leitner,F. *et al.* (2011) The protein-protein interaction tasks of BioCreative III: classification/ranking of articles and linking bio-ontology concepts to full text. *BMC Bioinformatics*, **12**(Suppl 8), S3.
5. Tanabe,L. and Wilbur,W.J. (2006) A priority model for named entities. *Proceedings of the BioNLP Workshop on Linking Natural Language and Biology (LNLBioNLP '06)*, 33-40.
6. Tanabe,L., Thom,L.H., Matten,W., *et al.* (2006) SemCat: semantically categorized entities for genomics. *AMIA Annual Symposium Proceedings*, 754-758.

# RLIMS-P: Literature-based curation of protein phosphorylation information

Manabu Torii<sup>1,2\*</sup>, Gang Li<sup>1,2</sup>, Zhiwen Li<sup>1,2</sup>, Irem Çelen<sup>1</sup>, Francesca Diella<sup>4</sup>, Rose Oughtred<sup>5</sup>, Cecilia Arighi<sup>1,2</sup>, Hongzhan Huang<sup>1,2</sup>, K. Vijay-Shanker<sup>2</sup>, Cathy H. Wu<sup>1,2,3</sup>

<sup>1</sup>Center for Bioinformatics and Computational Biology, University of Delaware, Newark, DE 19711, USA, <sup>2</sup>Department of Computer and Information Sciences, University of Delaware, Newark, DE 19711, USA, <sup>3</sup>Protein Information Resource, Department of Biochemistry, Molecular and Cellular Biology, Georgetown University Medical Center, Washington, DC 20007, USA, <sup>4</sup>EMBL, Germany, <sup>5</sup>Department of Genomics, Princeton University, Princeton, NJ, USA.

\*Corresponding author: Tel: 302 831 6162, E-mail: [torii@udel.edu](mailto:torii@udel.edu)

## Abstract

Annotation of protein phosphorylation information has been the focus of many biological knowledge bases. To support the literature-based curation of phosphorylation information, an information extraction (IE) system, named RLIMS-P, has been developed, which extracts protein phosphorylation information from biomedical literature. The system has been recently redesigned as RLIMS-P v2 and a new online curator website has been developed. The new website offers improvements for curation functionalities, including PubMed-style keyword search of extracted information, multiple views of retrieved information and their downloading, editing of automatically gathered information, and entity normalization. Curators from Phospho.ELM, Protein Ontology (PRO), and BioGrid were recruited to test the website in the BioCreative Track 5 - User Interactive Task (IAT). We expect the new website can be a useful tool for biocurators to search relevant literature and annotate phosphorylation information. Final results from the current evaluation test will be presented at the workshop.

## Introduction

The reversible phosphorylation of proteins is central to the regulation of most aspects of cell function. The flow of molecular information through signaling pathways frequently depends on protein phosphorylation mediated by specific kinases that recognize and phosphorylate specific sites in the target proteins (1). In many cases, deregulation of the kinase-substrate network has been linked to disease, including cancer (2). Given its relevancy, protein phosphorylation has been an active research area as well as the focus of curation in multiple knowledgebase, such as

Protein Ontology (PRO)<sup>1</sup>, PhosphoSitePlus<sup>2</sup>, Phospho.ELM<sup>3</sup>, and UniProt Knowledgebase (UniProtKB)<sup>4</sup>. To support review of relevant literature by biocurators, a rule-based information extraction (IE) system, named RLIMS-P, has been developed in our group (3,4). The system is designed to identify protein phosphorylation information reported in biomedical literature and it extracts entities involved in the phosphorylation event (kinase, substrate, and site). Recently the system has been revised as RLIMS-P v2 and applied to the entire MEDLINE. To make the large amount of information extracted from MEDLINE, a web interface for biocurators has also been redesigned. The new web interface allows users to search, retrieve, edit, and manage protein phosphorylation information online. In addition, we have integrated gene normalization results obtained with GenNorm (5) and the bibliography mapping information available in UniProtKB (6) in this web interface.

Based on the new interface design, we set up a curator website for BioCreative Track 5 - User Interactive Task (IAT). In this report, we describe this curator website, and introduce the data and the curation tasks considered for the IAT task.

## Material and Methods

### RLIMS-P system

RLIMS-P is a rule-based IE system designed to extract a kinase, a substrate, and a site that are involved in a phosphorylation event. The system consists of several text processing modules, including (i) a shallow parser that syntactically analyzes input sentences, (ii) a term classifier that identifies semantic categories of phrases, e.g., identification of protein names, (iii) a pattern-based IE engine that extracts entities involved in the target event, and (iv) an additional IE component that identifies an event reported across multiple sentences. This system has been recently redesigned as RLIMS-P v2 (7). One of the enhancements in the new system includes a design of the IE engine that eases management of extraction patterns. The new system can cover a large number of extraction patterns through combination of pattern fragments, instead of requiring a large set of complex patterns. Sentence simplification techniques in the original system, which improve pattern matching, were extended for the new design, based on the recent work in the group (8). RLIMS-P v2 was evaluated in different settings and F-scores for the extraction task were over 90%. For further information about RLIMS-P v2 and its evaluation results, readers may refer to (7).

---

<sup>1</sup> <http://pir.georgetown.edu/pro>

<sup>2</sup> <http://www.phosphosite.org>

<sup>3</sup> <http://phospho.elm.eu.org>

<sup>4</sup> <http://www.uniprot.org>

## The database

Phosphorylation information extracted from the MEDLINE archive using RLIMS v2 is stored in a database. Normalization of protein names obtained using GenNorm (5) is integrated in this database. In addition, the bibliography mapping service of PIR/UniProt is used to associate extracted information with UniProtKB entries. The resulting database is incrementally updated weekly in synch with MEDLINE citations in PubMed. The database initially built using the 2013 release of the MEDLINE archive contains phosphorylation information extracted from 165,840 abstracts, and links to 43,329 UniProtKB entries.

**1** Enter Keywords (accepts Boolean operators (AND, OR, NOT))  
Input keyword: "wnt signaling" [Submit Query] [Reset]

Or Enter PubMed IDs (PMIDs) delimited by "," or space, e.g., 15234272, 16436437.  
Input PMID: [Submit]

The latest 200 of 734 documents with potential phosphorylation are processed. Save PMIDs  
Documents RLIMS-P positive=177 where Kinase=44, Substrate=142 and Site=43  
Click here to see full results. Note the processing time may be long due to the big amount of PMIDs. ?

**2** View by PMID [Download]

Show Selected	PubMed ID	Protein Kinase	Phosphorylated Protein (Substrate)	Phosphorylation Site	No. of Sentences	Text Evidence/Curation
<input type="checkbox"/>	22369945	p21-activated kinase 1 ( pak1 ), protein kinase a	beta-catenin	Ser-675	1	
		pak1 k299r	beta-catenin	Thr-423	1	
		p21-activated kinase 1 ( pak1 )	beta-catenin	Ser-663	2	
<input type="checkbox"/>	22946057	ck1	p120-catenin	Ser, Thr	1	
		fyn, src	p120-catenin	Tyr	1	
		kinase d1 ( pkd1 )	beta catenin	Thr-120	2	

**3** Text Evidence [Back to Views] [Download] [Layout]

**4** Gene Normalization

Protein	Name	UniProtKB AC	Add UniProtKB AC	Annotation No.
Kinase	kinase d1 (pkd1)	P98161/PKD1_HUMAN ✓ X	<input type="checkbox"/>	1
	pkd1	P98161/PKD1_HUMAN ✓ X	<input type="checkbox"/>	1, 2
Substrate	beta-catenin	P35222/CTNBN1_HUMAN ✓ X	<input type="checkbox"/>	1, 3
	t120 beta-catenin	Not normalized	<input type="checkbox"/>	2

**5** RLIMS-P Annotation

No.	Kinase	Substrate	Site	Sentence	Comment	Validation
1	kinase d1 (pkd1)	beta catenin (beta-catenin)	Thr-120	4, 7		<input type="checkbox"/> ✓ X
2	kinase d1 (pkd1) (pkd1)	t120 beta-catenin	Thr-120	6		<input type="checkbox"/> ✓ X
3		beta catenin (beta-catenin)	Thr-120, Ser-37, Thr-41	5		<input type="checkbox"/> ✓ X

**6** Text Evidence

1 TI - Beta-catenin phosphorylated at threonine 120 antagonizes generation of active beta-catenin by spatial localization in trans-Golgi network.

2 AB - The stability and subcellular localization of beta-catenin , a protein that plays a major role in cell adhesion and proliferation , is tightly regulated by multiple signaling pathways .

3 While aberrant activation of beta-catenin signaling has been implicated in cancers , the biochemical identity of transcriptionally active beta-catenin ( ABC ) , commonly known as unphosphorylated serine 37 ( S37 ) and threonine 41 ( T41 ) beta-catenin , remains elusive .

4 Our current study demonstrates that ABC transcriptional activity is influenced by phosphorylation of T120 by Protein Kinase D1 ( PKD1 ) .

5 Whereas the nuclear beta-catenin from PKD1-low prostate cancer cell line C4-2 is unphosphorylated S37/T41/T120 with high transcription activity , the nuclear beta-catenin from PKD1-overexpressing C4-2 cells is highly phosphorylated at T120 , S37 and T41 with low transcription activity , implying that accumulation of nuclear beta-catenin alone cannot be simply used as a read-out for Wnt activation .

6 In human normal prostate tissue , the phosphorylated T120 beta-catenin is mainly localized to the trans-Golgi network ( TGN , 22/30 , 73% ) , and this pattern is significantly altered in prostate cancer ( 14/197 , 7.1% ) , which is consistent with known down regulation of PKD1 in prostate cancer .

7 These in vitro and in vivo data unveil a previously unrecognized post-translational modification of ABC through T120 phosphorylation by PKD1 , which alters subcellular localization and transcriptional activity of beta-catenin .

**Figure 1**-Snapshot of the main pages in RLIMS-P website; namely search, result table, and text evidence. 1-6 refer to the functionalities listed in the main text.

## **The web interface**

For BioCreative IV - IAT task, a new curator website has been set up ([http://research.bioinformatics.udel.edu/text\\_mining/rlimsp2/](http://research.bioinformatics.udel.edu/text_mining/rlimsp2/)). The website (Figure 1) supports the following functionalities:

1. Search and retrieval of phosphorylation information gathered by RLIMS-P using the PubMed-style query, as well as the query by PMIDs;
2. Display of a query result (a table of kinase, substrate, and site) with different ‘view’ options (e.g., group by kinase, substrate, or PMID) as well as sorting options;
3. Display of text evidence (MEDLINE abstracts with highlighted entities);
4. Provision of protein normalization information for kinases and substrates using GenNorm, a state-of-the-art normalization tool (5);
5. A user login for editing, saving and exporting curated annotations;
6. Downloading of phosphorylation information in the CSV format, and that of evidence text in the BioC format (9);
7. Support of different browsers: Google Chrome, Mozilla Firefox, Internet Explorer 9, and Safari.

These functionalities as well as the usage of the website are described in a help document ([http://research.bioinformatics.udel.edu/text\\_mining/rlimsp2/files/RLIMSP\\_help.pdf](http://research.bioinformatics.udel.edu/text_mining/rlimsp2/files/RLIMSP_help.pdf)). The website has been developed with the help of PRO curators, but it is intended for a broader curation community, not limited to PRO curation.

## **The BioCuration task**

Three curators were recruited to test the RLIMS-P website. They are domain experts with experience on kinase-substrate event annotation, specifically annotation for Phospho.ELM, PRO, and PhosphoGRID/BioGRID databases. Curation guidelines were developed and they describe which entities should be captured (kinase, substrate and site) and how they should be normalized, along with exercises to get familiar with the curation criteria and the interface ([http://research.bioinformatics.udel.edu/text\\_mining/rlimsp2/files/RLIMSP\\_guidelines.pdf](http://research.bioinformatics.udel.edu/text_mining/rlimsp2/files/RLIMSP_guidelines.pdf)). The curation task requested is summarized below:

1. Given a set of 50 PMIDs, fill in the tuples of kinase, substrate and site with normalization information. Perform this task on a half of this collection using the curator website and on the other half without using it. The curator records the annotation results along with UniProtKB identifiers where possible. The curator will record the time spent.
2. Complete the user survey (<http://ir.cis.udel.edu/biocreative/survey2.html>).

All the annotation results will be reviewed by a senior PRO curator and the performance of the RLIMS-P system will be measured using standard performance measures, such as precision and recall. We should also examine the time spent for the manual curation and that for the RLIMS-P-assisted curation.

## **The datasets**

Three datasets tailored to the participating curators were prepared as below.

### *Dataset 1*

The first dataset was prepared for the PhosphoGRID/BioGRID curator. This dataset includes articles with phosphorylation information on yeast, published between 2012 and 2013. The selection of 50 PMIDs was based on the PubMed query: ("2012/01/01"[Date - Publication] : "3000"[Date - Publication]) AND (saccharomyces OR yeast) AND phosphory\*. The retrieved results were inspected to confirm that the contents were appropriate for the curation task.

### *Dataset 2*

The second dataset was prepared for the Phospho.ELM curator. This dataset was compiled for any kinase-substrates relation reported in articles published in 2013. The selection of 50 PMIDs were based on the PubMed query: ("2013/01/01"[Date - Publication] : "3000"[Date - Publication]) AND kinase AND phosphory\*. Again, the retrieved results were inspected so that the contents were appropriate.

### *Dataset 3*

The third dataset was prepared for the PRO curator. This set contained a subset of abstracts from Dataset 1 (11) and a subset from Dataset 2 (36), and the remaining ones were collected from literature related to transient potential receptors (TRP).

## **Results and Discussion**

The original RLIMS-P system had been used for PRO curation of protein forms (10,11), Phospho.ELM curation (12), and pathway curation (13). It had also been used to provide information for another text-mining system, eFIP, which extracts functional impact of phosphorylation events (14,15). We expect that the enhancements in RLIMS-P v2 and the new curator website described in this report can further help curators to annotate phosphorylation information or a text mining tool based on RLISM-P to extract biomedical knowledge. Final results from the current evaluation test will be closely examined and our analyses as well as the obtained results will be presented at the workshop.

## **Funding**

This work was supported by the National Science Foundation [ABI-1062520 to M.T., C.A., H.H., K.V., and C.H.] and the National Library of Medicine of the National Institutes of Health [G08LM010720 to C.A., H.H., K.V., and C.H.].

The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

## Acknowledgement

We acknowledge the work of Drs. Narayanaswamy and Ravikumar, who played an integral role in developing the original RLIMS-P system. We would like to thank the IAT Track organizers Drs. Phoebe Roberts, Sherry Mathis, and Catalina Oana Tudor for their assistance during RLIMS-P evaluation.

**Conflict of Interest:** none declared.

## References

1. Pawson, T. and M. Kofler. 2009. Kinome signaling through regulated protein-protein interactions in normal and cancer cells. *Curr Opin Cell Biol* 21:147-153.
2. Zhang, L. and R.J. Daly. 2012. Targeting the human kinome for cancer therapy: current perspectives. *Crit Rev Oncog* 17:233-246.
3. Narayanaswamy, M., K.E. Ravikumar, and K. Vijay-Shanker. 2005. Beyond the clause: extraction of phosphorylation information from medline abstracts. *Bioinformatics (Oxford, England)* 21 Suppl 1:i319.
4. Hu, Z.Z., M. Narayanaswamy, K.E. Ravikumar, K. Vijay-Shanker, and C.H. Wu. 2005. Literature mining and database annotation of protein phosphorylation using a rule-based system. *Bioinformatics (Oxford, England)* 21:2759.
5. Wei, C.-H. and H.-Y. Kao. 2011. Cross-species gene normalization by species inference. *BMC bioinformatics* 12 Suppl 8.
6. The UniProt Consortium. 2012. Reorganizing the protein space at the Universal Protein Resource (UniProt). *Nucleic acids research* 40:D71.
7. Torii, M., C.N. Arighi, Q. Wang, C.H. Wu, and K. Vijay-Shanker. 2013. Text Mining of Protein Phosphorylation Information Using a Generalizable Rule-Based Approach, ACM Conference on Bioinformatics, Computational Biology and Biomedical Informatics (ACM-BCB) 2013.
8. Peng, Y., Tudor, C.O., Torii, M., Wu, C. H. and Vijay-Shanker, K. 2012. iSimp: A sentence simplification system for biomedical text, IEEE International Conference on Bioinformatics and Biomedicine (BIBM 2012), Philadelphia, USA.
9. Comeau, D.C., R. Islamaj Dogan, P. Ciccarese, K.B. Cohen, M. Krallinger, F. Leitner, Z. Lu, Y. Peng, et al. 2013. BioC: a minimalist approach to interoperability for biomedical text processing. *Database (Oxford)* 2013:bat064.
10. Ross, K.E., C.N. Arighi, J. Ren, H. Huang, and C.H. Wu. 2013. Construction of Protein Phosphorylation Networks by Data Mining, Text Mining, and Ontology Integration: Analysis of the Spindle Checkpoint. *in press*.
11. Ross, K.E., C.N. Arighi, J. Ren, D.A. Natale, H. Huang, and C.H. Wu. 2013. Use of the protein ontology for multi-faceted analysis of biological processes: a case study of the spindle checkpoint. *Frontiers in genetics* 4.
12. Dinkel, H., C. Chica, A. Via, C.M. Gould, L.J. Jensen, T.J. Gibson, and F. Diella. 2011. Phospho.ELM: a database of phosphorylation sites--update 2011. *Nucleic acids research* 39:D261.

13. Schmidt, C.J., L. Sun, C.N. Arighi, K. Decker, K. Vijay-Shanker, M. Torii, C.O. Tudor, C. Wu, and P. D'Eustachio. 2012. Pathway curation: Application of text-mining tools eGIFT and RLIMS-P, p. 523. IEEE.
14. Arighi, C.N., A.Y. Siu, C.O. Tudor, J.A. Nchoutmboube, C.H. Wu, and V.K. Shanker. 2011. eFIP: a tool for mining functional impact of phosphorylation from literature. *Methods in molecular biology* (Clifton, N.J.) *694*:63.
15. Tudor, C.O., C.N. Arighi, Q. Wang, C.H. Wu, and K. Vijay-Shanker. 2012. The eFIP system for text mining of protein interaction networks of phosphorylated proteins. *Database* (Oxford) *2012*:bas044.

# Egas – Collaborative Biomedical Annotation as a Service

David Campos<sup>1\*</sup>, Joni Lourenço<sup>1</sup>, Tiago Nunes<sup>1</sup>, Rui Vitorino<sup>2</sup>, Pedro Domingues<sup>2</sup>, Sérgio Matos<sup>1\*</sup> and José Luís Oliveira<sup>1</sup>

<sup>1</sup>IEETA/DETI, University of Aveiro, Campus Universitário de Santiago, 3810-193 Aveiro, Portugal

<sup>2</sup>Chemistry Department, University of Aveiro, Campus Universitário de Santiago, 3810-193 Aveiro, Portugal

\*Corresponding author: Tel: +351 234 370 500, E-mails: {david.campos,aleixomatos}@ua.pt

## Abstract

In this paper we present Egas, a web-based platform for biomedical text mining and collaborative curation. The web tool allows users to annotate texts with concept occurrences as well as with relations between concepts. Annotations may be imported together with the documents using one of the accepted input formats, or may be added during the annotation process, either manually or by calling a document annotation service. Users can further inspect, correct or remove automatic text mining results, add new annotations, and export the results to standard formats.

The tool is available at <http://bioinformatics.ua.pt/egas> and is compatible with most recent versions of Google Chrome, Mozilla Firefox, Internet Explorer and Safari.

## Introduction

Egas (**Figure 1**) is a web-based platform for biomedical text mining and collaborative curation. It allows users to annotate texts with occurrences of concepts and relations between these concepts. The annotation tool follows what we termed an “annotation-as-a-service” paradigm. Document collections, users, configurations, annotations, back-end data storage, as well as the tools for document processing and text mining, are all managed centrally. This way, a curation team can use the service, configured according to their requisites, taking advantage of a centrally managed pipeline.

The tool is based on the idea of *Projects*. A *Project* consists of a curation or document annotation task, performed on a collection of documents, by a team of (one or more) curators, and considering a pre-defined set of concept and relation types defined by the curation guidelines. A project administrator is responsible for managing the users (curators) associated to the project and the project characteristics, such as annotation guidelines and target concepts and relations. Projects may be public or private, in which case they are only accessible by users that have been added by the project manager.

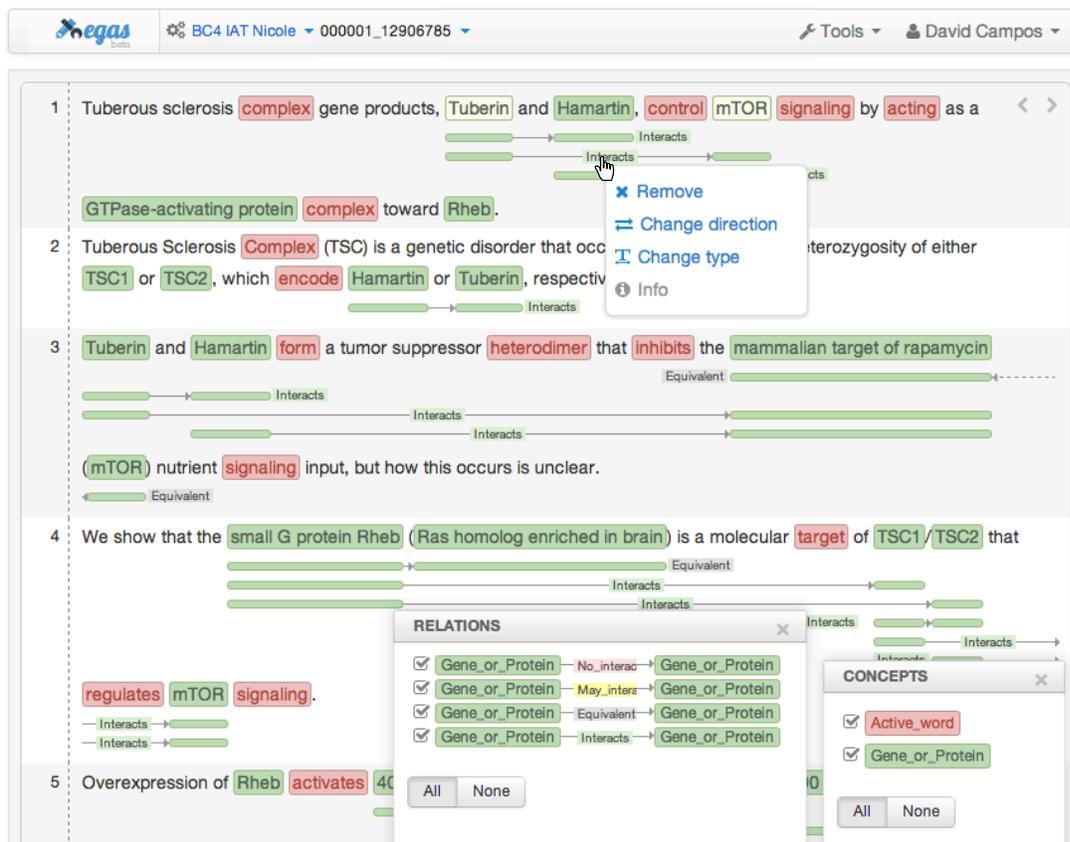


Figure 1: Egas main user interface.

To create the document collection for a project, three import options are available:

- Local – from a local collection of documents;
- Remote – from remote resources using a list of identifiers;
- Search – from remote resources by submitting search queries.

For each Project, the project administrator can freely define the relevant concept and relation types, according to the requisites of the task. To facilitate the annotation work, each different concept and relation type is associated with a markup color. A relation type is defined by specifying the types of the intervening concepts and assigning a name to the relation. For example, for protein-protein interactions, after defining a concept type “Protein”, an “Interacts” relation can be defined between two concepts of type “Protein”.

Curators can start from the raw text and add the concept and relation annotations as they review the documents, or they can start from preprocessed texts, containing automatically identified concepts and relations that they will revise. This can be achieved by importing a previously processed document collection or by using integrated concept and/or relation extraction services to pre-process a set of documents in the collection.

## System Description

### Project management

Project management allows project administrators to specify configurations of the annotation task, such as:

- Project: manage project information and annotation guidelines;
- Users: manage curators;
- Concepts: manage concepts to annotate;
- Relations: manage relations to annotate;

Through the project panel, the administrator can indicate which curators are allowed to annotate the documents associated with the project. Moreover, the project administrator can also provide a brief description of the annotation task, and upload documents describing the guidelines of the annotation task, which are accessible to the curators associated with the project.

Concept and relation management allows adding or removing target concept types and defining relations between those concepts, as well as selecting the associated markup colors.

### Adding documents to a project

Importing documents from the client machine supports RAW, A1 [1] and BioC [2] formats. Regarding the A1 format, if corresponding annotation files are provided, both the text of the documents and any concept and relation annotations are imported to the database. Importing documents from remote resources supports both PubMed and PubMed Central, through a list of identifiers. The corresponding documents are loaded from the remote resource and displayed to the users, so they can select which ones to import. Likewise, users can submit a query to search either PubMed or PubMed Central. In this case, the search results are obtained from the remote resource and displayed to the users for selection.

### Exporting project documents

Exporting project documents to an external resource supports both A1 and BioC formats. Such feature allows users to store the generated information locally, in order to add it to a local knowledge base or for using in text mining pipelines, for instance.

### Automatic concept and relation mining

Egas provides an interface that allows using external automatic annotation tools that are available as web-services. For instance, users can automatically annotate a document with specific concepts and respective relations, and then correct the provided annotations. It currently integrates an automatic service for protein-protein interactions (PPIs) annotation, providing the following annotations: *a*) protein concepts; *b*) relations between proteins (PPIs); *c*) relations marking equivalent protein mentions (e.g. acronyms and long forms); and *d*) active words that

may indicate the presence of PPIs. The service is implemented on top of Neji [3], using Gimli [4] to perform machine learning-based protein name recognition. BioThesaurus [5] is used to normalize recognized names, through the application of prioritized dictionary matching [3]. Equivalent protein relations are added using a simple abbreviation resolution technique, and PPIs are recognized through a rule-based approach using dependency-parsing trees.

### **Annotation interface**

Figure 1 shows the tool's main user interface. The central box displays the content of the text being curated, showing the concepts and relations that have been identified. Concepts are shown as colored boxes, using the colors defined in the project configuration. Relations are shown as lines, tagged with the relation type. The colored boxes connected by the relation markup are placed under the concepts that participate in the relation, and are colored with the same color as the respective concept, making it easy to identify the entire relation. Moreover, hovering the cursor over the relation markup also highlights the involved concepts.

The boxes on the lower right corner allow curators to select the concept and relation types they want to appear highlighted in the text.

During the curation task, concepts and relations can be added, edited or removed. Hovering the mouse over an annotation shows the corresponding semantic type and, by right-clicking, a menu opens that allows removing the annotation. A new concept annotation is added by selecting a text span in the annotation window. This opens a concept type selection box for choosing the concept type for the new annotation.

To add a relation, the user clicks the first concept in the relation while pressing the “Alt” key, and then clicks the second concept also while pressing the “Alt” key. Relations are considered directional, so the order in which the concepts are added to a relation is important. For example, if a relation “promotes” is defined, the order needs to be considered. As for concepts, right-clicking over an existing relation allows removing it. In the same menu, the user can easily change the relation type and/or direction (Figure 1).

### **Implementation**

Text-processing modules, such as the concept and relation annotation services, were implemented in Java, the article fetching modules were also built in Java, and the web interface was developed using HTML5, CSS3, and JavaScript, in order to allow fast processing of large documents and support mobile devices. The resulting information, such as annotations and relations, is stored in a relational database. Finally, all database operations are performed using secured RESTful web-services, allowing easy integration with mobile devices, such as smartphones and tablets.

## Case Study – the BioCreative IV IAT task

The proposed curation task consists in the identification and extraction of biomolecular events described over PubMed abstracts related to neuropathological disorders, including protein-protein interactions, protein expression and post-translational modifications. To create the corpus for this task, a collection consisting of more than 135 thousand PubMed abstracts was first obtained with the PubMed search:

```
“Neurodegenerative Diseases”[MeSH Terms] OR “Hereditary Degenerative Disorders, Nervous System”[MeSH Terms] AND hasabstract[text] AND English[lang]
```

The documents were then ranked according to their relevance for extracting protein-protein interactions, using a SVM classifier trained on the BioCreative III PPI Article Classification Task data [6]. Finally, the top-ranked 100 documents were selected for the task.

Four curators were selected, and each was assigned 50 documents from the corpus to curate. Curators were asked to annotate 25 of their assigned documents using the available PPI annotation service described above, and the remaining 25 documents without using this service, in order to assess its impact on curation effort. In the first case, curators had to revise the automatically generated annotations, correcting any erroneous concept or relation annotation and adding missing ones. In the second case, curators had to annotated all mentions of protein names and all protein interactions described in each document. The tool recorded the time taken by each curator to curate each document, as well as the number of annotated concepts and relations.

## Conclusion

A tool for collaborative document annotation and curation is proposed. The tool allows teams of curators to work on a shared curation project, following a set of configurable concept and relation types. The curation task can be performed over a collection of raw text documents or by reviewing automatic concept and relation annotations, obtained either with the included concept and relation identification service or through external annotation tools. Documents can be imported in raw text, A1 and BioC formats, and the final annotations may be exported to A1 and BioC formats. Apart from the local import option, it is also possible to create a document collection by importing from PubMed and PubMed Central either through a list of identifiers or by submitting a search to these services.

We are currently working on adding real-time collaboration features, providing instant feedback of users' interactions within a document. Thus, multiple users can change a document at the same time, showing exactly who changed what. A project chat will also be available, allowing users to discuss details of the annotation task.

The tool is available at <http://bioinformatics.ua.pt/egas> and is in active development by the Bioinformatics group at the University of Aveiro, Portugal, aiming to provide an annotation-as-a-service solution through a flexible, configurable and user-friendly environment.

## Funding

This work was supported by FEDER through the Operational Program Competitiveness Factors - COMPETE and by National Funds through FCT - Foundation for Science and Technology, in the context of the projects FCOMP-01-0124-FEDER-010029 (FCT reference PTDC/EIA-CCO/100541/2008), FCOMP-01-0124-FEDER-022682 (FCT reference PEst-C/EEI/UI0127/2011) and Incentivo/EEI/UI0127/2013. S. Matos is funded by FCT under the Ciência2007 programme.

## References

1. Standoff format - brat rapid annotation tool [<http://brat.nlplab.org/standoff.html>].
2. Comeau DC, Islamaj Doğan R, Ciccarese P, Cohen KB, Krallinger M, Leitner F, Lu Z, Peng Y, Rinaldi F, Torii M, Valencia A, Verspoor K, Wiegers TC, Wu CH, Wilbur WJ: BioC: a minimalist approach to interoperability for biomedical text processing. *Database (Oxford)* 2013, 2013:bat064.
3. Campos D, Matos S, Oliveira JL: A modular framework for biomedical concept recognition. *BMC Bioinformatics* 2013, 14:281.
4. Campos D, Matos S, Oliveira J: Gimli: open source and high-performance biomedical name recognition. *BMC Bioinformatics* 2013, 14:54.
5. Liu H, Hu ZZ, Zhang J, Wu C: BioThesaurus: a web-based thesaurus of protein and gene names. *Bioinformatics* 2006, 22:103–105.
6. Krallinger M, Vazquez M, Leitner F: The Protein-Protein Interaction tasks of BioCreative III: classification/ranking of articles and linking bio-ontology concepts to full text. *BMC Bioinformatics* 2011, 12:S3.

# *tagtog*: Interactive Human and Machine Annotation of Gene Mentions in PLOS Full-Text Articles

Juan Miguel Cejuela, Peter McQuilton<sup>1\*</sup>, Laura Ponting<sup>1</sup>, Steven J. Marygold<sup>1</sup>, Raymund Stefancsik<sup>1</sup>, Gillian H. Millburn<sup>1</sup>, Burkhard Rost<sup>2</sup> and the FlyBase Consortium<sup>+</sup>.

Paul-Hindemith-Allee 6, 80939, Munich, Germany; <sup>1</sup>Department of Genetics, University of Cambridge, Downing Street, Cambridge CB2 3EH, UK; Bioinformatics-i12, <sup>2</sup>Department of Informatics, Technical University of Munich (TUM), 85748 Garching, Germany;

To whom correspondence should be addressed: pam51@gen.cam.ac.uk

<sup>+</sup>The Current FlyBase Consortium comprises: William Gelbart, Nicholas H. Brown, Thomas Kaufman, Kathy Matthews, Maggie Werner-Washburne, Richard Cripps, Kris Broll, Madeline Crosby, Gilberto dos Santos, David Emmert, L. Sian Gramates, Kathleen Falls, Beverley B. Matthews, Susan Russo, Andrew Schroeder, Susan E. St. Pierre, Pinglei Zhou, Mark Zytkevich, Boris Adryan, Helen Attrill, Marta Costa, Steven Marygold, Peter McQuilton, Gillian Millburn, Laura Ponting, Raymund Stefancsik, Susan Tweedie, Josh Goodman, Gary Grumbling, Victor Strelets, Jim Thurmond and Harriett Platero.

## Abstract

We present the *tagtog* system, a web-based annotation framework that can be used to mark-up biological entities and concepts in full-text articles. *tagtog* leverages user manual annotations in combination with automatic machine-learned annotations to provide accurate gene symbol and name identification in biomedical literature. For this submission we present, in collaboration with the FlyBase database curation team at Cambridge University, the task of identifying and extracting mentions of *Drosophila melanogaster* gene symbols and names in full-text biomedical articles from the PLOS stable of journals. We show here the results of three experiments with different sized corpora, and assess gene recognition performance. Finally, we would like to extend an invitation for Biocurators at the BioCreative IV Track 5 -- User Interactive Task (IAT) to come and find us to try *tagtog* themselves.

## Introduction

*tagtog* (<http://tagtog.net>) is a web-based framework for the annotation of named entities. A user creates a project, defines a named-entity recognition task (such as gene annotation), and uploads a set of text documents to the system. Each document is then displayed in a web editor where the user can add, delete, or correct the information relevant to the annotation task. An example of the user interface is shown in Figure 1. The user can add the annotation of an entity by selecting the corresponding word(s) and remove it by clicking another time on the selection. During the

course of their work, a biocurator will need to analyse thousands or even millions of documents, an undertaking that is impossible to do through manual means alone. To address this problem, the tagtog system leverages machine-learning methods to perform the same type of annotations computationally. Initially, the tool is trained with a small set of manually annotated documents. Once trained, tagtog can be used to process a set of novel documents wherein automatically generated predictions are made that then can be reviewed and corrected by the user. It is this continuous and interactive re-training of the machine learning methods with user feedback, that can lead to an ever-improving performance in automatic prediction. Once optimized, the trained machine learning methods can be used to process and annotate a large volume of documents to a sufficiently accurate level. We envisage this process will save curator time and effort and lead to significant increases in gene annotation efficiency. Finally, the annotated documents can be exported (in *anndoc* XML format) for the particular user's application needs, such as for further curation or manipulation into curation records for database submission.

The screenshot shows the tagtog web interface. At the top, there is a search bar with the text 'tagtog' and a search icon. To the right of the search bar are links for 'jmcejuela', 'Help', and 'Log out'. Below the search bar is a navigation menu with links for 'Guidelines', 'Corpus', 'Learning', 'Downloads', and 'Admin'. The main content area is divided into two sections. On the left, there is an 'Upload' button and a list of items: 'pool' and 'gold'. On the right, there is an 'Entities Tally' section with the following data: '# total entities: 379', '# uniq. entities: 5', and a list of items: 'LINT: 90/90', 'dCoREST: 33/33', 'dL(3)mbt: 117/117', 'dLint-1: 131/131', and 'l(3)mbt: 8/8'. The main content area displays a document titled 'LINT, a Novel dL(3)mbt-Containing Complex, Represses Malignant Brain Tumour Signature Genes'. Below the title is an 'Abstract' section with the following text: 'Mutations in the l(3)mbt tumour suppressor result in overproliferation of Drosophila larval brains. Recently, the derepression of different gene classes in l(3)mbt mutants was shown to be causal for transformation. However, the molecular mechanisms of dL(3)mbt-mediated gene repression are not understood. Here, we identify LINT, the major dL(3)mbt complex of Drosophila. LINT has three core subunits—dL(3)mbt, dCoREST, and dLint-1—and is expressed in cell lines, embryos, and larval brain. Using genome-wide CHIP-Seq analysis, we show that dLint-1 binds close to the TSS of tumour-relevant target genes. Depletion of the LINT core subunits results in derepression of these genes. By contrast, histone deacetylase, histone methylase, and histone demethylase activities are not required to maintain repression. Our results support a direct role of LINT in the repression of brain tumour-relevant target genes by restricting promoter access.'

**Figure 1** – Example of the document display and editor in tagtog.

In the following section, we describe the system's current features and technical details. In the final section, we will describe our three experiments to date, using papers from the FlyBase (<http://flybase.org>) bibliography.

## Current Features

- **Browser support:** The system runs on all major current browsers only requiring HTML5 and javascript. Chrome and Firefox are officially supported. Other browsers like Opera, Safari, and Internet Explorer (9 and 10) are regularly tested but lack official support. Access to the system can be provided using our invitation system.

- **Multiple projects:** Users can create different annotation projects and load their own dictionaries and corpora.
- **Multiple users:** Multiple users on the same project are also supported, allowing curation teams to view and annotate the same set of papers.
- **Multiple entities:** Support for the annotation of multi-entities in the same project is also desired and expected to be ready by the end of 2013.
- **Active learning:** tagtog actively asks for user feedback on predicted annotations. A proposed mechanism was already developed in an early version of tagtog, presented last year at the BioCreative 2012 workshop.
- **Document searching:** Papers can be searched using the search tool at the top of the interface. Options include searching by document ID (based on the DOI), entities, or whether a paper has been fully annotated or not. In the future we hope to add the facility to search by PubMed ID (PMID).
- **Import options:** Any paper following the NCBI Journal Publishing Tag Set<sup>1</sup> or the BioMed Central format<sup>2</sup> can be uploaded to tagtog. This includes full-text papers from the PLOS, BioMed Central, Chemistry Central, and Springer Open collections. In the near future, we will accept papers from the new JATS format<sup>3</sup> and plain text files.
- **Export options:** Annotations for each paper can be exported as a tab-separated list of terms linked to PMIDs. In addition, *anndoc* XML for entire corpora can be downloaded, allowing users to archive annotations. We hope to allow soon export of annotations in the new BioC format<sup>5</sup>.

### Defining the Annotation Guidelines

Upon project creation, the first step for a user is to define the annotation guidelines (Figure 2). These define what and how to annotate the project documents. The user currently has the following options:

- **Entity:** choose the entity class name to annotate.
- **Entity Dictionary:** upload a user-defined dictionary/ontology of collected entity names. The dictionary can contain synonyms and database-specific IDs, allowing data integrity checks and seamless integration of the results with the parent database.
- **Meta Information:** define a list of checkboxes for document triage, e.g., whether the article contain disease mentions, or information on a new transgene.
- **Annotables:** select the sections of the full-text articles that can be annotated (and trained with). The annotation of figures' and images' captions is decided independently: *always*, *never*, or *section-dependent*.
- **Pre-Annotations:** if activated, upon entity selection or de-selection, all instances of an annotation will be 'pre-annotated' or automatically de-selected. These pre-annotations require user confirmation before final annotation.

## Example

Guidelines    Corpus    Learning    Downloads    Admin

Entity  
Entity Dictionary  
Meta Information  
Annotables  
Pre-Annotations

### Annotable Sections

Select those document sections you want to annotate:

- Title
- Abstract
- Introduction & Background
- Materials & Methods
- Results
- Conclusion & Discussion

Annotate Figures & Tables

Figure 2 - Annotation guidelines.

### Defining a Corpus

All full-text XML documents that follow the NCBI Journal Publishing Tag Set format (versions 2.x and 3.0) (for example all PLOS journals) and the BioMed Central format (for example, all articles from BioMed Central, Chemistry Central, and SpringerOpen) can be uploaded either as single documents or in batches. The system's internal parser recognizes the documents' sections, subsections, figures, tables, and some additional meta-information such as the paper's original URL. The project corpus can be augmented progressively as the user sees fit. Currently, documents are placed in two folders, the *pool* folder, where training documents are placed, and the *gold* folder, where a smaller set of previously annotated documents are used for the evaluation of the machine learning methods' performance (these documents are never used for training).

### Downloading Your Data

The user can export the entire corpus upon request. Documents are annotated in XML documents with an in-house-defined format, called *anndoc*. The current version of *anndoc* (0.3) supports in-line entity annotations (without being nested), meta information, hierarchy of sections, figures, tables, and their captions. *Anndoc* follows a HTML-like format, with a header for meta

information and a body for the document's text and annotations. The tags have been so selected to match those of HTML5's for an easy display of the document in current browsers. In addition, the annotations for each document can also be exported as a tab-separated list of terms linked to the corresponding PMID. In the future, we are keen to adopt other standard output formats if there is sufficient user demand.

### **The Machine Learning Component of tagtog**

A core, defining characteristic of the system is that the users can choose the entity type to annotate, such as gene, GO term, or disease. The system boasts a general-purpose named-entity recognizer. The recognizer is customized to the prediction task at hand by means of user feedback and by the dictionary/ontology of entity terms. The system is configured to allow expansion of annotation types with new machine annotators via plugins to enable annotation of multiple data-types and languages.

If desired, the machine-learning component of tagtog can be turned off to allow biocurators to use the tagtog interface exclusively for manual curation.

## **Interactive Annotation Task**

### **FlyBase Gene Mention Annotation Task**

In collaboration with the FlyBase database<sup>4</sup> (<http://flybase.org>) at Cambridge University, we have undertaken the task of identifying and extracting mentions of genes of the drosophila genus (fruit flies) in full-text biomedical articles. Since September 2012, FlyBase have been testing tagtog for its adoption in their Genetic Literature Curation pipeline. Currently, FlyBase has two well-defined application cases for tagtog:

- **Gene to publication links:** To use tagtog to automatically generate gene-publication links. These will be used to populate the FlyBase gene report and reference report web pages, as well as to pre-populate the community curation FlyBase Fast-Track-Your-Paper tool and FlyBase curation records.
- **Skim Curation:** In addition to the gene-to-publication links mentioned above, FlyBase curators could use the metadata information tags to triage publications for further curation. This would replicate the 'skim curation' step in the FlyBase paper curation pipeline.

In this task, we will show how FlyBase could integrate tagtog with their curation pipeline, address specific practical details such as how to consume the generated output formats, what is the subjective user experience with the tool on a daily basis, and what short-comings and improvements can be identified. In addition, during this process we will expand the FlyBase-tagtog corpus with more manually annotated documents and assess the performance of the machine learning methods. We hope to release this corpus for use by other text-mining groups as

we believe the corpus to be the largest and most complete gene mention annotation set in full-text articles currently available.

To date, FlyBase curators have manually annotated 431 papers using the tagtog interface. All papers are full-text PLOS journal articles from between 2011 and 2013. The following document sections were annotated: title, abstract, results, materials and methods, and figure and table legends. The paper annotations have been used to iteratively train the machine-learning component of tagtog.

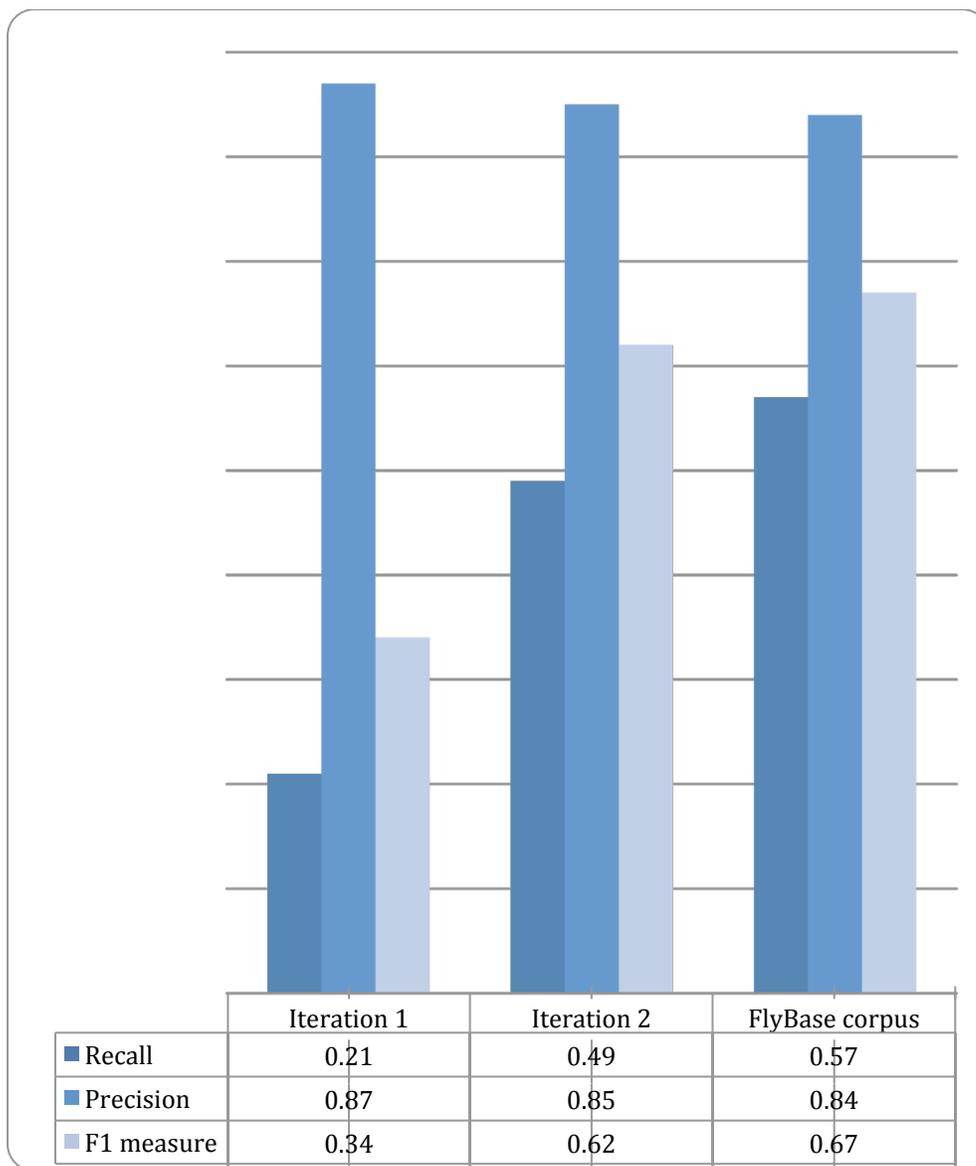
So far, we have performed three annotation+benchmark iterations. In the first two iterations, annotations were done partially manually by a sole curator (Peter McQuilton) and automatically by the system. In the third iteration, all five FlyBase curators annotated papers manually. All the manual annotations and corrections were performed using tagtog's document editor interface.

The document sets for the three iterations are as follows:

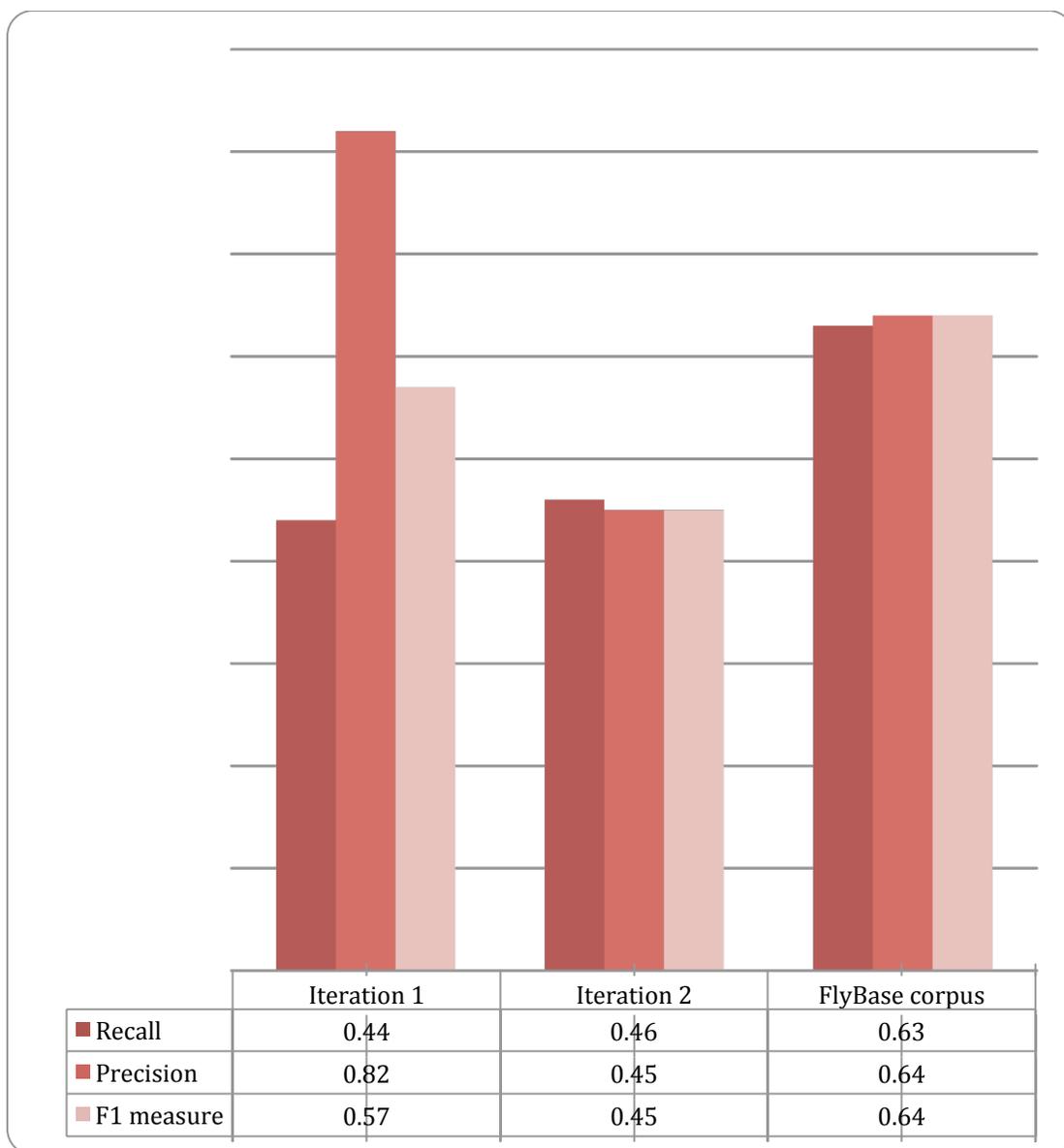
- **Iteration 1:** A sole curator (P. McQuilton) manually annotated a training set of 20 articles. Trained with these documents, the system was applied to predict an unlabelled test set of 99 articles. The curator then went through the test set and, corrected, added, or removed the predicted annotations when appropriate. In the end, mismatched annotations between the original predictions and the revised annotations were counted as errors.
- **Iteration 2:** the previous two sets were united to form a training set of 119 articles. For evaluation, the user manually annotated a test set of 20 articles (which we will refer to as the ‘Gold Standard’). The system was trained on the 119 articles and benchmarked against the 20 test articles. In contrast to Iteration 1, in this case prediction errors could be read off directly from mismatches against the test set.
- **Iteration 3:** FlyBase jamboree combined set. The previous two sets, plus a further 312 papers curated by 5 different FlyBase curators, were combined to form an annotated corpus of 431 Fly-related papers. These papers were used to train tagtog before assessment on the Gold Standard set.

For the performance benchmarks, we used standard evaluation measures for named-entity recognition (NER), namely: precision (P), recall (R), and F1-Measure (F1). Only exact matches between the predictions and the test annotations were counted as correct. That is, the predictions had to match the same exact word boundaries. Two types of counts were considered: 1) unique entities on a document basis. That is, for a test entity X in a document, the predictions were right if at least one mention of that entity could be identified in that document, wrong otherwise. Equivalently, all unique entities identified by the predictions but not present on the test annotations were counted as errors. 2) All entity mentions for all documents. That is, for all entity mentions, matching predictions and test annotations were counted as correct. Mismatched mentions, either false positives or false negatives, were counted as errors. Note that for testing, only the annotable sections defined by the curator are compared.

Figure 3 shows the entity recognition performance for all entity mentions in a paper, that is, the ability of tagtog to identify the presence of a gene mention, either as a symbol or name. The figure shows that the performance has steadily improved in proportion to the corpus size. The same performance improvement behavior is seen for unique entity recognition (figure 4), that is, the ability to identify the presence of a gene mention at least once in a paper.



**Figure 3** - Entity recognition performance over all three corpora sizes. Iteration 1 was using a corpus of 20 documents to train tagtog to identify gene symbols/names in 99 documents. Iteration 2 used a training set of 119 documents to assess performance on a ‘gold standard’ set of 20 papers. Iteration 3 used the FlyBase corpus, composed of 431 articles, to train tagtog, with performance assessed on the gold standard set.



**Figure 4** – Entity instance recognition performance over all three corpora sizes. Iteration 1 was using a corpus of 20 documents to train tagtog to identify gene symbols/names in 99 documents. Iteration 2 used a training set of 119 documents to assess performance on a ‘gold standard’ set of 20 papers. Iteration 3 used the FlyBase corpus, composed of 431 articles, to train tagtog, with performance assessed on the gold standard set.

### BioCreative IAT challenge

In addition to the experiments mentioned above, we are recruiting other biocurators from outside FlyBase to participate in the testing and assessment of tagtog. In this challenge, biocurators will be first asked to manually annotate mentions of an entity class in up to 20 papers using tagtog, and then asked to annotate a second set of 20 papers that have been marked up by the machine-learning component of tagtog, trained on the first set of 20 papers. Note that the entity class to

annotate will be defined by the biocurators. Both annotation tasks will be timed, enabling judgement not only of the accuracy of the tagtog annotation but also the time cost/benefit of these assisted annotations.

## Conclusions

Although overall a moderate performance, we believe these early evaluation results to be promising. To our knowledge, these results represent one of the first NER evaluations with a substantial amount of full-text articles in the biomedical field. NER with full-text articles is understood to be considerably more difficult than for abstracts, a focus scope that was more studied in the past<sup>7,8</sup>. It must be noted that the machine learning method is not specialized to the problem, except for the ontology of terms given by the user. In particular, we have not specialized the machine learning to focus on the genes of a single organism, *Drosophila melanogaster*. On the whole, prediction performance appears to increase with an increase in the volume of training data. The continuous learning of tagtog is designed to generate cheaper (in terms of manual curation effort) training data, by taking advantage of semi-automatic annotation. Still, a central goal for the curation task proposed in this submission, is to further improve the performance of the tagtog system so it operates at a sufficiently accurate level that it can be incorporated into the FlyBase literature curation pipeline.

## Author Contributions

JMC and PM devised the experiments and wrote the manuscript. PM led the curation process and gave directions as for the development of the user interface. PM, SM, LP, RS, and GM annotated the corpus. JMC develops tagtog.

## Funding

This work was supported by private funding for Juan Miguel Cejuela and the National Human Genome Research Institute at the National Institutes of Health (P41 HG00739), and the Medical Research Council (UK) (G1000968) for members of FlyBase.

## References

1. Journal Publishing Tag Set. URL <http://dtd.nlm.nih.gov/publishing/>.
2. BioMed Central format. URL <http://www.biomedcentral.com/about/xml>.
3. PLOS. URL <http://www.plos.org/>.
4. Journal Article Tag Suite. URL <http://jats.nlm.nih.gov/>
5. Comeau, R.I.D. et al., *BioC: A Minimalist Approach to Interoperability for Biomedical Text Processing*. DATABASE, 2013 submitted.
6. P.McQuilton, and the FlyBase Consortium. Opportunities for text mining in the FlyBase genetic literature curation workflow. Database (Oxford), 2012:bas039, 2012.

7. Larry Smith, Lorraine Tanabe, Rie Ando, Cheng J. Kuo, Fang I. Chung, Chun N. Hsu, Yu S. Lin, Roman Klinger, Christoph Friedrich, Kuzman Ganchev, Manabu Torii, Hong-fang Liu, Barry Haddow, Craig Struble, Richard Povinelli, Andreas Vlachos, William Baumgartner, Lawrence Hunter, Bob Carpenter, Richard Tsai, Hong J. Dai, Feng Liu, Yifei Chen, Chengjie Sun, Sophia Katrenko, Pieter Adriaans, Christian Blaschke, Rafael Torres, Mariana Neves, Preslav Nakov, Anna Divoli, Manuel M. Lopez, Jacinto Mata, and John W. Wilbur. Overview of BioCreative II gene mention recognition. *Genome Biology*, 9(Suppl 2), 2008.
8. Zhiyong Lu, Hung Y. Kao, Chih H. Wei, Minlie Huang, Jingchen Liu, Cheng J. Kuo, Chun N. Hsu, Richard Tsai, Hong J. Dai, Naoaki Okazaki, Han C. Cho, Martin Gerner, Illes Solt, Shashank Agarwal, Feifan Liu, Dina Vishnyakova, Patrick Ruch, Martin Romacker, Fabio Rinaldi, Sanmitra Bhattacharya, Padmini Srinivasan, Hongfang Liu, Manabu Torii, Sergio Matos, David Campos, Karin Verspoor, Kevin Livingston, and W. Wilbur. The gene normalization task in BioCreative III. *BMC Bioinformatics*, 12(Suppl 8):S2+, 2011. ISSN 1471-2105. URL <http://dx.doi.org/10.1186/1471-2105-12-S8-S2>.

# Customisable Curation Workflows in Argo

Rafal Rak\*, Riza Batista-Navarro, Andrew Rowley, Jacob Carter and Sophia Ananiadou

National Centre for Text Mining, University of Manchester, UK

\*Corresponding author: Tel: +441613063090, E-mail: rafal.rak@manchester.ac.uk

## Abstract

We adapt Argo, a Web-based text mining workbench, to accommodate a curation task for BioCreative IV Track 5 User Interactive Task. The system allows users to build and run custom processing workflows composed of a subset of available elementary processing components. Running such workflows results in the generation of annotations for each input document. Argo has the capability to include a manual annotation editor as part of any workflow and therefore facilitates manual inspection and correction of otherwise automatically produced annotations. As a case study, we propose a curation task that involves the annotation and identification (normalisation) of concepts relevant to metabolic processes, including chemicals, gene or gene products, and trigger/bioprocess expressions.

**Keywords:** Curation, Automatic annotation, Metabolic processes, Workflows, UIMA

## Introduction

Argo<sup>1</sup> is a Web-based workbench for collaborative development and evaluation of text-processing workflows (1). The workbench features an ever growing library of elementary processing components, developed mostly at the National Centre for Text Mining (NaCTeM). They range from simple data (de)serialisation to syntactic and semantic analysis. The major features and the principal requirements of the ongoing development of Argo are: 1) the ease of combining elementary text-processing components to form meaningful and comprehensive processing workflows; 2) the ability to manually intervene in the otherwise automatic process of annotation by correcting or creating new annotations; and 3) user collaboration by providing sharing capabilities for user-owned resources.

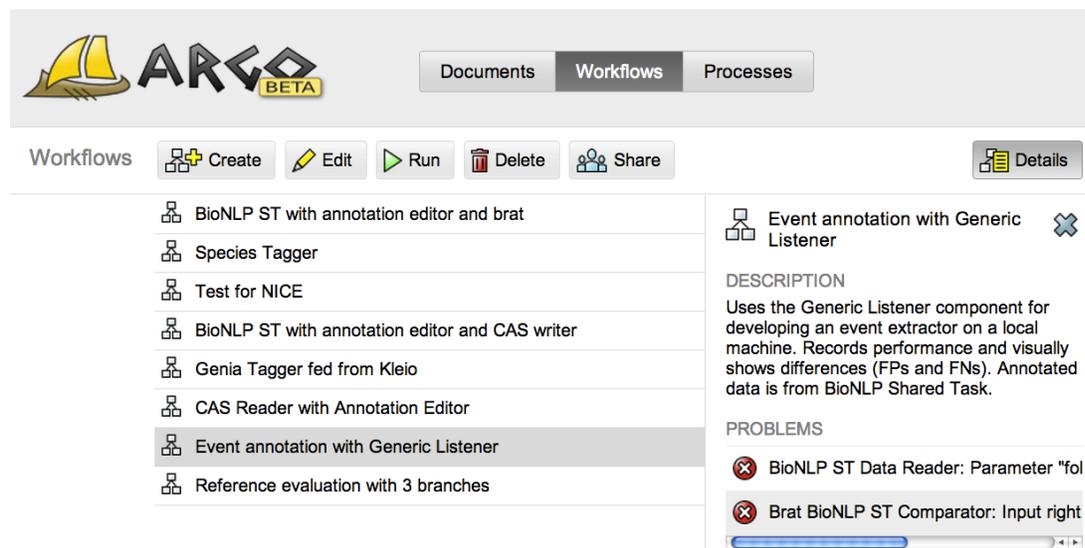
As such, Argo is a generic platform, meant to accommodate a variety of tasks and domains. The tasks in Argo are defined by creating *workflows*, i.e., arranging several elementary processing components by interconnecting their outputs and inputs and setting up their configuration parameters. The generic flow involves 1) reading source data, 2) performing automatic and/or manual annotation on the data, and 3) saving the annotations (usually together with the source data). Each of the three steps is highly configurable by choosing elementary processing

---

<sup>1</sup> <http://argo.nactem.ac.uk>

components that best fit the task at hand. For example, reading source data may be accomplished by deserialising user-uploaded files (in various formats) as well as by fetching data from remote Web services, such as search engines.

In the following sections we describe the capabilities of Argo in the context of the proposed curation task, which involves the annotation of concepts relevant to metabolic processes.



**Figure 1.** Screenshot of the Argo application. The view shows the Workflows panel that lists and enables managing user-created workflows.

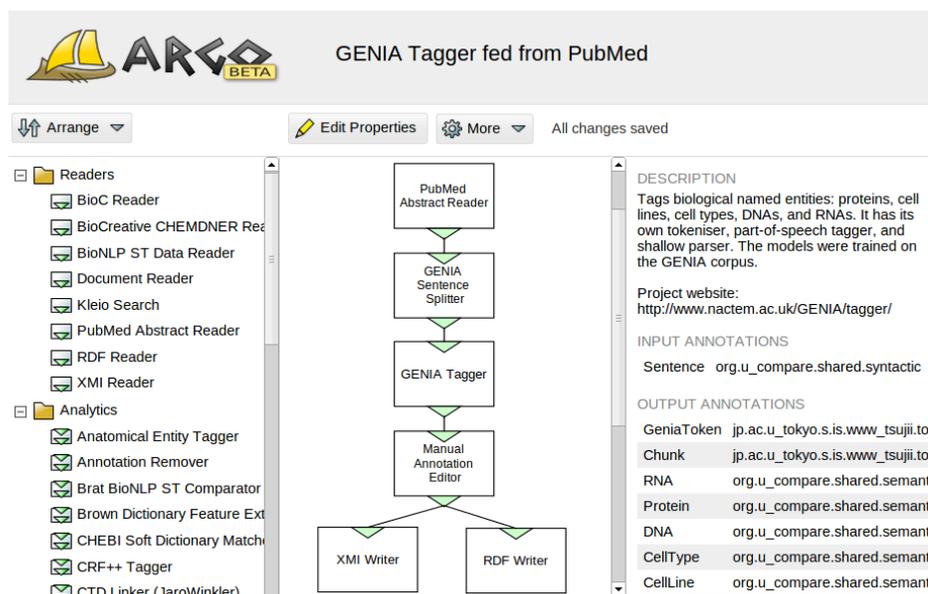
## System Overview

The user interface of Argo is split into three panels, Documents, Workflows, and Processes, as shown in Figure 1.

The *Documents* panel serves to manage user-owned files. These are usually text documents uploaded by the user to be used as the source of workflow processing. They may also be the product of executing workflows, e.g., XML files containing annotations, available for the user to download. User-created workflows are managed in the *Workflows* panel. This panel also serves to initiate the processing of workflows whose progress may be tracked in the *Processes* panel (see below).

Users create or edit workflows with the use of a graphical diagram editor shown in Figure 2. Components (represented as blocks) are linked together to form connected graphs—workflows. Although the most common arrangement is a pipeline (where the output of each component is connected to at most one input of another component), Argo also accommodates complex arrangements with multiple branching and merging points as shown in the figure.

Argo supports and is based on the Unstructured Information Management Architecture (UIMA) (2). The architecture is an OASIS standard<sup>2</sup> for ensuring interoperability of individual processing components by defining common data structures and interfaces. Therefore, the platform has the potential of running any UIMA-compatible processing component.



**Figure 2.** Argo's workflow diagramming interface. The left-hand-side panel lists available processing components; the middle panel is the actual diagram editor (the blocks represent components); and the right-hand-side panel displays contextual information.

### System Curation Features

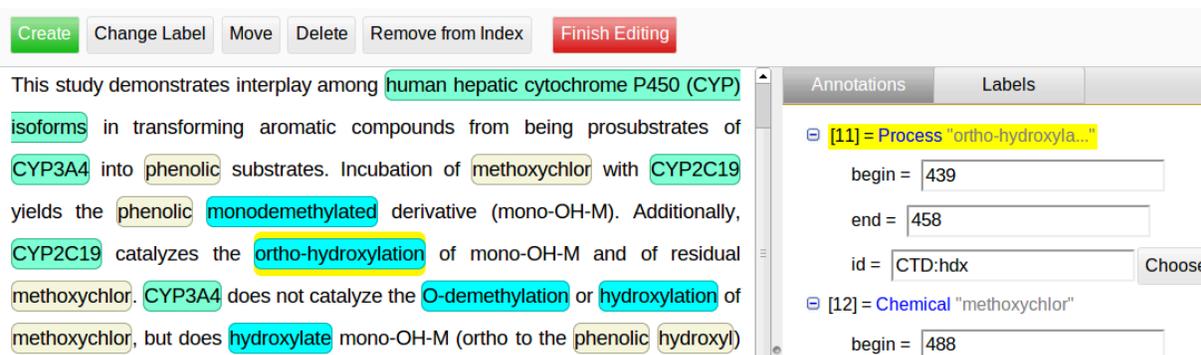
One of the novel features of Argo is the ability to manually intervene in the processing of a workflow by the use of user-interactive processing components. An example of this type of component is the *Manual Annotation Editor*, which is part of the workflow shown in Figure 2. The featured workflow works as follows: PubMed Abstract Reader fetches PubMed abstracts matching given PubMed IDs (set as the component's configuration parameter). The abstracts are then passed onto GENIA Sentence Splitter that annotates sentence boundaries. Farther in the pipeline, GENIA Tagger performs multiple tasks including tokenisation, part-of-speech tagging, chunking, and finally, named entity recognition. The latter three use the sentence boundary information extracted in the GENIA Sentence Splitter. The name entities include categorisation into RNA, protein, DNA, cell type and cell line.

The automatic processing pauses at the Manual Annotation Editor component. At this point, the user has the option to open the Manual Annotation Editor's user interface and modify or delete

<sup>2</sup> <http://www.oasis-open.org/committees/uima>

existing, automatically extracted annotations or add new ones. Figure shows a fragment of the Manual Annotation Editor's interface. The features of the editor include:

- selecting a document for annotation from a list of available (previously processed) documents;
- removing or re-assigning labels to already (automatically) annotated spans of text;
- adding new annotations by selecting a span of text and assigning a label to it from a set of available labels (the central panel in Figure 3);
- adding new or deleting existing non-span-of-text annotations (meta annotations);
- editing annotation features (attributes) organised in an expandable tree structure (the right-hand-side panel in Figure 3);
- support for overlapping (intersecting) annotations (see Figure 4);
- a GUI for assigning identifiers from external databases (Figure 5); and
- filtering by label.



**Figure 3.** Annotation Editor interface. The central panel allows for manual on-text manipulation of annotations, whereas the right-hand-side panel allows for the more refined edition of annotation attributes.



**Figure 4.** Overlapping (intersecting) annotations are visibly distinct in the annotation editor.

### Supported Data Formats

Due to its modular nature, Argo supports a variety of input and output data formats. Formats are thus selected by choosing appropriate data reading and writing components from the library of available processing components. The following lists a selection of Argo (de)serialisation components that are of interest to biological curation tasks.

- *Pubmed Abstract Reader* fetches abstracts directly from PubMed using a list of PubMed IDs as a parameter.
- *Kleio Search* remotely fetches PubMed abstracts matching a query set as a parameter.
- *BioC Reader* and *BioC Writer* deserialise/serialise selected annotations from/into BioC format<sup>3</sup>.

**Figure 5.** Linking annotations to identifiers in external databases, e.g., ChEBI.

- *RDF Reader* and *RDF Writer* deserialise/serialise text and annotation from/into RDF. Serialised annotation RDF graphs may then be reused in other applications, e.g., in query engines supporting SPARQL. SPARQL may also be used to add, delete or modify annotations (3), which, in turn, can be read back into Argo.
- *BioNLP ST Data Reader* deserialises triple files (containing plain text, standoff annotations of named entities, and standoff annotations of events or structured relationships) as defined in the BioNLP Shared Task<sup>4</sup>.
- *Agreement Evaluator* analyses two or more input annotation efforts (coming from different branches in a workflow) and produces a tab-separated file that reports agreement rates between the inputs. It may serve to compute inter-annotator agreement.

<sup>3</sup> <http://bioc.sourceforge.net>

<sup>4</sup> <http://2013.bionlp-st.org>

A common approach is to end an Argo workflow with a XMI Writer, since the data in this format may be later read back into Argo with XMI Reader and translated into other formats.

## Annotation of metabolic processes

Our proposed curation task involves the annotation and identification (normalisation) of concepts relevant to metabolic processes.

### Background

Metabolic reactions or processes are the building blocks of metabolic pathways, which have received little attention from the biomedical NLP community compared to signalling pathways (4). Whilst the latter are centred on protein-protein or ligand-receptor interactions, metabolic pathways primarily consist of a series of biochemical reactions. For this task, the curators will be asked to annotate named entities and action terms relevant to metabolic processes.

We define metabolic process by taking the definition from the interaction types ontology in the Comparative Toxicogenomic Database<sup>5</sup> (CTD), i.e., “the biochemical alteration of a molecule’s structure, excluding changes in expression, stability, folding, localization, splicing and transport”.

### The Task

The task involves the annotation of chemical compounds (CCs), genes or gene products (GGPs) and expressions signifying a metabolic process (triggers). Both CCs and GGPs may play the role of reactant (entity undergoing the alteration), product (entity into which the reactant is changed) or modifier (entity driving the alteration) in a metabolic process. Each metabolic process is signified by an action term which is a span of text that best expresses the process in text. It can be a verb, verb nominalisation, adjective or adverb, e.g., *phosphorylation*, *generates*, *acetylated*. Additionally, a unique identifier from external resources will be assigned to each of the above-mentioned annotations. The resources are ChEBI<sup>6</sup> for CCs, UniProt<sup>7</sup> for GGPs and CTD for action words. Detailed annotation guidelines, examples and instructions were provided for curators<sup>8</sup>.

### Input

The input of the annotation process is a set of 60 PubMed abstracts, tagged in the CTD as relevant to different types of metabolic processes. Half of the set is used for manual curation and the other half for automatic annotation.

---

<sup>5</sup> <http://ctdbase.org>

<sup>6</sup> <http://www.ebi.ac.uk/chebi>

<sup>7</sup> <http://www.uniprot.org>

<sup>8</sup> <http://argo.nactem.ac.uk/annotating-metabolic-processes>

## Annotation process

The following workflows were prepared for the curators:

1. *Manual annotation*: This workflow reads PubMed abstracts and opens a Manual Annotation Editor for a curator to tag the entities of interest without any support from automatic processing available in Argo.
  - Input: PubMed abstract identifiers.
  - Output: Annotation files in XMI (interchangeable) format.
2. *Automatic annotation*: This workflow reads PubMed abstracts and performs automatic recognition of the entities of interest. This workflow is purely automatic and does not involve any manual intervention from the curators. The objective of this workflow is to automatically “pre-annotate” the input abstracts for later manual inspection. It consists of the following semantic analysis components: GENIA Tagger (5) and a refactored version of OSCAR 3 (6) as concept recognisers, and string similarity-based concept linkers to ChEBI, UniProt and CTD.
  - Input: PubMed abstract identifiers.
  - Output: Annotation files in XMI format.
3. *Manual correction*: This workflow reads files that already contain annotations (coming from the second, automatic workflow) and opens a Manual Annotation Editor for a curator to correct (remove, add, modify) the automatically recognised annotations.
  - Input: Annotation files in XMI format.
  - Output: Annotation files in XMI format.

Although the second and third workflows could be combined into a single workflow, for the purpose of evaluation they are defined separately. The three workflows are publicly available in Argo.

## Output

For each PubMed abstract, a set of annotated text spans corresponding to GGPs, CEs and triggers will be returned as output. The annotations for each text span include the location (i.e., document offsets), semantic label (any of GGP, CE and trigger) and the corresponding unique identifier from external resources. The annotations will be available in XMI, BioC, and RDF formats upon the completion of the task by curators.

## Evaluation

To evaluate the efficiency of using Argo, the curators are asked to compare the time spent on purely manual annotation (i.e., using only the *Manual annotation* workflow described above) against the time spent on the correction of automatically pre-annotated abstracts (i.e., using *Manual correction* after the *Automatic annotation* workflow). Since both annotation tasks are performed in Argo, it is, in fact, the evaluation of the automatic support available in Argo.

To evaluate effectiveness, we will measure the performance of the automatic pre-annotation stage using dedicated components in Argo (e.g., *Reference Evaluator*) that report on the precision, recall and F-score against the manual annotations.

## System Performance

The performance of Argo is ultimately defined by the contents of user-created workflows. Here we report on the performance of the processing components featured in Argo that are of interest to the proposed curation task. Table 1 shows the precision, recall, and F-score of GENIA Tagger and OSCAR 3.

Tool	Precision	Recall	F-score	Data set
GENIA Tagger	67.45	75.78	71.37	NLPBA 2004 (7)
OSCAR 3	90.31	79.29	84.44	Sciborg corpus (8)

**Table 1.** Performance of GENIA Tagger and OSCAR 3.

Argo has previously been used in the annotation of a corpus of 231 full-text *Marine Drugs* journal articles with compound-target interaction information<sup>9</sup>. After the names of marine drugs and enzymes in the documents were automatically pre-annotated by a named entity recogniser, two NaCTeM-supervised domain experts used Argo's Manual Annotation Editor to review the automatically annotated named entities and mark up interactions between them. A sample of 30 fully-annotated documents revealed that each one contained an average of 124 automatically recognised entities, of which an average of only 3.2 entities were removed (as being false positives), 1.2 were modified (i.e., annotation boundaries extended or narrowed down) and 5 were added by the annotators.

## Funding

This work was partially supported by Europe PubMed Central funders (led by Wellcome Trust).

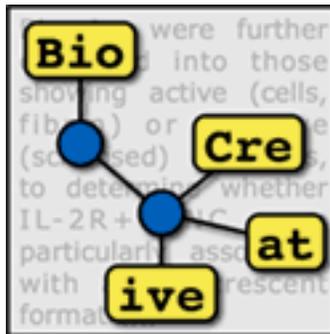
## References

1. Rak, R., et al., *Argo: an integrative, interactive, text mining-based workbench supporting curation*. Database, 2012. **2012**.
2. Ferrucci, D. and A. Lally, *UIMA: an architectural approach to unstructured information processing in the corporate research environment*. Nat. Lang. Eng., 2004. **10**(3-4): p. 327-348.
3. Rak, R. and S. Ananiadou. *Making UIMA Truly Interoperable with SPARQL*. In Proceedings: *7th Linguistic Annotation Workshop and Interoperability with Discourse*. 2013. Sofia, Bulgaria.

---

<sup>9</sup> The project outcome report is in preparation.

4. Li, C., M. Liakata, and D. Rebolz-Schuhmann, *Biological network extraction from scientific literature: state of the art and challenges*. Briefings Bioinf., 2013.
5. Tsuruoka, Y., et al. *Developing a Robust Part-of-Speech Tagger for Biomedical Text*. In Proceedings: *Advances in Informatics - 10th Panhellenic Conference on Informatics*. 2005. Springer-Verlag.
6. Kolluru, B., et al., *Using Workflows to Explore and Optimise Named Entity Recognition for Chemistry*. PLoS ONE, 2011. **6**(5): p. e20181.
7. Kim, J.-D., et al. *Introduction to the bio-entity recognition task at JNLPBA*. In Proceedings: *International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*. 2004. Geneva, Switzerland: Association for Computational Linguistics.
8. Copestake, A., S. Teufel, and B. Waldron. *Flexible Interfaces in the Application of Language Technology to an eScience Corpus*. In Proceedings: *4th UK E-Science All Hands Meeting*. 2006.



## METAGENOMICS PANEL

### Organizers:

- Lynette Hirschman, MITRE
- Cathy H. Wu, University of Delaware

### Invited Speaker:

- Evangelos Pafilis, Institute of Marine Biology Biotechnology and Aquaculture  
Hellenic Centre for Marine Research (HCMR)

### Panelists:

- Jim Cole, Michigan State
- George Garrity, Names for Life
- Nikos Kyrpides, JGI
- Folker Meyer, Argonne National Laboratory
- Tatiana Tatusova, NCBI

## Overview

The BioCreative workshop brings together developers of text mining/natural language processing systems, to apply these systems to challenge problems defined by the curators of various biological databases and/or researchers who need to search the biological literature. We are starting a new track for BioCreative to explore the text mining needs of the metagenomics community. Our plan is to proceed in several stages:

1. A panel discussion at BioCreative IV (Oct. 7-9, 2013)
2. Participation at the BioCuration Conference (Toronto, April 2014) related to curation and metagenomics – the exact form is to be determined.
3. A challenge evaluation track at BioCreative V (fall 2015) using one or more data sets of relevance to the metagenomics community.

The goal of this panel then is to familiarize the BioCreative text mining community with the text mining needs of the metagenomics community; to expose this community to metagenomics practices, resources, and text mining needs; to identify some candidate tasks and data sets for a challenge evaluation at BioCreative V (2015); and to identify possible organizers for a metagenomics challenge evaluation task for BioCreative V.

### Background on BioCreative

To date, the BioCreative challenge evaluations have addressed a variety of problems provided by users and curators of biological databases and the biomedical literature, including:

- Triage techniques to prioritize articles for curation for a specific biological database, e.g., articles containing chemical/gene/disease information; or articles containing experimental evidence for protein-protein interactions, or experimental evidence for gene function for genes of a specific model organism.
- Extraction of curatable information for typical biocuration tasks, including protein-protein interaction, gene function and/or basic biological entity extraction for genes, proteins, or chemicals mentioned in the biological literature; these tasks have often included mapping the entities to their unique biological identifiers (e.g., EntrezGene id) and identification of evidence passages in the full text of articles.
- Demonstration of interactive systems to assist curation, for example, by finding gene and protein mentions, linking these to the appropriate EntrezGene or Uniprot identifiers, and making these available for interactive human review and correction.

## **Metagenomics Text Mining Needs**

The text mining and curation needs of the metagenomics community are somewhat different from the needs that the BioCreative participants have been exposed to so far. In part, this is because the curation activities are different: curation for the metagenomics community is much closer to the experimental data. A major focus has been the development of analysis pipelines and repositories for whole genome and metagenomics data (e.g., KBase, MG-Rast, GOLD; also RDP and BioSample).

A major challenge for metagenomics is the capture of the relevant metadata (e.g., isolation source, time and location of sample collection, etc.). There is relatively less curation from the literature, although there is certainly some interest in capturing these kinds of data from the literature and from semi-structured metadata found in the repositories. This could be done retrospectively (after paper publication or data deposit), or it could be done prospectively, through development of interactive tools to link unstructured descriptions or names with the appropriate identifiers or controlled vocabulary terms or ontological concepts.

Examples include taxon identifiers for species in a metagenomics sample, or use of EnvO for habitat/isolation source information. George Garrity's Names for Life is a good example of an existing suite of tools for linking names to the appropriate nomenclature. Evangelos Pafilis is developing tools and resources for species and environments; he will be giving a talk on his work before the DOE panel session. Another application that we have discussed is description of microbial phenotypes.

## **Panel Materials and Organization**

This initial panel will help to identify possible organizers and data sets for the BioCreative V metagenomics track and to explore which text mining groups would be interested in participating in such a track. We will also be putting in place a Metagenomics User Advisory Group to guide this process, which would include the panelists plus other interested users from the metagenomics community (suggestions are welcome).

The panel will consist of a brief introduction from the organizers, followed by short statements from the panelists. This will take up the first 45 minutes. We will then open the floor to questions and discussion. Finally, we will set aside a few minutes at the end to summarize candidate tasks, determine which tasks would be of interest to current BioCreative participants and identify next steps.

## **Evangelos Pafilis, PhD**

Research Fellow

Institute of Marine Biology Biotechnology and Aquaculture

Hellenic Centre for Marine Research (HCMR)

P.O. Box 2214

Heraklion, 71003

Crete, Greece

pafilis@hcmr.gr, <http://epafilis.info>

### **Research interests as connected to metagenomics and text mining**

I am a Research Fellow at the Hellenic Center for Marine Research, Crete, Greece. Coming from a background in data integration and literature mining in the biomedical domain, I am exploring how such techniques could be adapted to serve a broader range of biological research fields. My overall aim is to support answering questions that span the whole spectrum of biological organization from genes all the way up to ecosystems.

My current focus is in the recognition of organism names and environment descriptors mentioned in text. Such term recognition could help linking biological entities, such as species and sequences, to their environmental context; a valuable piece of input for fields like metagenomics and biodiversity research.

To this end I am experimenting with publicly available literature and data collections, I am involved in the development of corpora necessary for the evaluation of such methods, and working closely with molecular biologists, ecologists, and computational biologists to integrate the literature mining derived information with input from sequence similarity, ecology and other types of data analyses.

### **Priorities for text mining to improve data capture for metagenomics research**

The continuous community efforts in developing standards such as MiXS to capture genomic/metagenomic and other type of sequence contextual information highlight [a] the importance of the latter.

To facilitate the standard-compliant reporting of contextual data, checklists with detailed guidelines are being developed [1]. Necessary ontological resources such the Environment Ontology (EnvO) [b] are also subject of active research to the same end.

Sequence contextual information can be found as free-text in metadata fields like sample source, sampling method, location and time.

Named entity recognition and interactive curation could play an active role in assisting sequence contextual information annotation even further e.g. by processing such pieces of free text and suggesting relevant ontology terms [2] and or link to records in commonly used resources.

BioCreative's experience in organizing relevant evaluation challenges could well drive relevant developments to improve data capture for metagenomics.

On a technical level, the identification of entity types such as environment descriptive terms, organism mentions, anatomy terms, geographical locations is required. Essential to this end, is the establishment of evaluation strategies including the required test datasets and gold-standard corpora.

Text-mining assisted data capture could occur at different stages. One possible point is to support researchers upon sequence submission. Another is by assisting database annotators while curating a set of sequences. Both call for the development of user-friendly submission/curation tools that would suggest fine-grained, accurate and standard-compliant annotations.

Besides named entity recognition and interactive curation, text mining could be of use at extracting important relations such as species – environment associations from sequence and literature repositories. Tools that support functional annotation and small chemical molecule recognition could enrich the extracted facts of knowledge even further.

Last but not least, equally important to employing text mining to improve data capture, is the adoption of the text mining tools by metagenomics analysis pipelines. The integration of text-mining derived information (e.g. via the interactive curation and/or via predictions for mis-annotated sequences) could extend the functionality offered to biologists. This would not only increase the outreach of text mining tools, but also demonstrate the importance of endorsing the standards and related tools.

### **Technologies needed**

Parts of the previously mentioned requirements can be addressed by existing tools and resources. Taxonomic name recognition has been long studied. NamesforLife [c], LINNAEUS [3], SPECIES and ORGANISMS [4], are all relevant applications.

Ontologies could play a key role both as name sources and by supporting inferences based on their hierarchy. EnvO, for example, could provide names environment descriptors such as biomes, habitats, environmental features, materials and conditions. ENVIRONMENTS [d] is a dictionary-based supporting the identification of EnvO terms in text.

Components from other fields could also be of potential use. BioGeomancer, for example, is a tool from the Biodiversity field to convert localities mentioned in text into a geospatial descriptions [5].

Augmented browsing tools such that the Names4Life [c], Reflect [6], and the MegxBar [e] browser extensions, provide ideas on how possible interactive curation interfaces could look like.

The tasks performed by such tools could be improved even further by a relevant evaluation challenge. BioCreative could play a pivot role by prioritizing entity recognition tasks, facilitating their evaluations, and orchestrating the communication to the metagenomics community.

An important issue to bear in mind is the dynamic nature of key underlying resources. Taxonomies are evolving, and so do ontologies. Thus, being always up-to-date is a key issue.

Finally, hands on workshops, “Hackathons”, involving experts from both sides could support exchange of experience and drive software development towards specific tasks even further.

### **Candidate challenge problems**

According to the above to improve data capture for metagenomics research a combined effort in named entity recognition and interactive curation is required. Standard precision and recall analysis for the identification of entity types of interest to the metagenomics community, coupled with demonstrations of interactive curation applications could be evaluated drawing on the experience from previous BioCreative challenges [7].

The existence of relevant corpora, such as the Linnaeus-100 [3] and the Species-800 [4] (for species name identification) and the one, under preparation, in the context of the ENVIRONMENTS-EOL [f] project (for environment descriptive term identification), could be of potential use.

### **Web Resources**

- a. GSC – MiXS publications: [http://gensc.org/gc\\_wiki/index.php/GSC\\_Publications](http://gensc.org/gc_wiki/index.php/GSC_Publications)
- b. The Environment Ontology: <http://environmentontology.org/>
- c. The NamesforLife Technology: <http://services.namesforlife.com/about>
- d. The ENVIRONMENTS Tagger: <http://environments.hcmr.gr>
- e. The MegxBar Browser Extension: <http://www.microb3.eu/work-packages/wp5>
- f. The ENVIRONMENTS-EOL Project: <http://environments-eol.blogspot.com>

## References

1. Yilmaz P, Gilbert JA, Knight R, Amaral-Zettler L, Karsch-Mizrachi I, et al. (2011) The genomic standards consortium: bringing standards to life for microbial ecology. *The ISME journal* 5: 1565–1567.
2. Hirschman L, Clark C, Cohen KB, Mardis S, Luciano J, et al. (2008) Habitat-Lite: a GSC case study based on free text terms for environmental metadata. *Omics: a journal of integrative biology* 12: 129–136.
- Gerner M, Nenadic G, Bergman CM (2010) LINNAEUS: a species name identification system for biomedical literature. *BMC bioinformatics* 11: 85.
3. Pafilis E, Frankild SP, Fanini L, Faulwetter S, Pavloudi C, et al. (2013) The SPECIES and ORGANISMS Resources for Fast and Accurate Identification of Taxonomic Names in Text. *PLoS one* 8: e65390.
4. Guralnick, R. P., Wieczorek, J., Beaman, R., & Hijmans, R. J. (2006). BioGeomancer: automated georeferencing to map the world's biodiversity data. *PLoS biology*, 4(11), e381.  
doi:10.1371/journal.pbio.0040381
5. Pafilis E, O'Donoghue SI, Jensen LJ, Horn H, Kuhn M, et al. (2009) Reflect: augmented browsing for the life scientist. *Nature Biotechnology* 27: 508–510.
6. Wu CH, Arighi CN, Cohen KB, Hirschman L, Krallinger M, et al. (2012) BioCreative-2012 virtual issue. *Database : the journal of biological databases and curation* 2012: bas049.

## **James R. Cole, Ph.D.**

Assistant Professor

Center for Microbial Ecology

Michigan State University

567 Wilson Rd - Rm 2225A

East Lansing, MI 48824

+1 (517) 410-0589

Our group runs several "boutique" sequences databases. The RDP Ribosomal Database Project offers aligned and annotated rRNA sequences, along with specialized analysis tools to the research community. RDP data and tools are utilized in fields as diverse as human health, taxonomy and phylogenetics, microbial ecology, environmental microbiology, and nucleic acid chemistry. Our RDP FunGene project performs a similar function for ecofunctional genes, genes that serve as markers for specific ecologically important enzymatic pathways, for example carbon and nitrogen cycling pathways, important sources and sinks of greenhouse gases, and antibiotic resistance markers, important in infectious disease. Much of the use of our databases comes from researchers interested in gene-targeted metagenomic studies. These studies target rRNA or ecofunctional genes either by PCR amplification of the gene target or by detection and assembly of targeted genes from shotgun metagenomic data. Researchers use our databases to develop oligonucleotide primers and probes, to estimate the diversity expected in a given gene, and as reference sequences to compare with their unknowns.

As a secondary database, we compile all of our sequence data and much of our annotation from the INSDC databases so we have become familiar with annotation issues in these databases. For example, the INSDC feature table defines an rRNA feature key, which we take advantage of when we search for rRNA sequences. However the /product qualifier is free text, and of very little use in determining the type of rRNA gene. This is an example of where an enforced controlled vocabulary would be incredibly useful; there are only a few types of rRNA gene products. For our RDP FunGene database, we find it best to ignore the /gene and /product qualifiers on the first pass. The major issue in this case is not just the use of nonstandard names, but the uncertain reliability of the assignments. For many genes, current tools that annotate by sequence similarity still mis-identify many ORFs, especially between closely related paralogs. Use of evidence codes in the INSDC records is sparse and so we often rely on small hand-annotated sequence sets we receive from researchers to guide a general similarity search strategy, the results of which we refine through protein trees and other means.

It would be great if researchers could be convinced to make better use of existing annotation standards, such as the MixS series of minimal information standards from the Genomic Standards Consortium. However, the incentives that would catalyze this just aren't in place. A

more realistic approach might be using improved text mining systems with the ability to correctly infer evidence codes. Systems capable of intelligently traversing references to find the important papers could be helpful, since the publications most relevant to experimental evidence might not be listed directly in the INSDC sequence records. Of course, there are likely many existing text mining tools that would help with some of our tasks, and I look forward to learning more about these.

## George M. Garrity, Sc.D.

Professor, Microbiology and Molecular Genetics, Michigan State University, East Lansing, MI  
Co-founder and Managing Member, NamesforLife, LLC, East Lansing, MI

### Overview of current relevant efforts

Extraction of biological information from the primary literature is a non-trivial task. In addition to the hidden biases introduced by curators (who must interpret results described by authors in a variety of ways) the process is confounded by a “chicken or egg” dilemma. Proper storage of curated data requires a well-designed data model *a priori*, but the knowledge of how to properly model (normalize) the data is only available *posteriori*, once the structure of the underlying information is well understood. Initial assumptions about what information can be extracted and how that information should be stored must undergo near continual revision as additional resources are added to a project corpus. How might one maintain quality, consistency and usability of stored observational data over time, knowing that both the information and the underlying data are fluid and often inconsistent or even contradictory?

We have developed a generalized process to mitigate these challenges that includes a flexible data model, document analysis methods, and a workflow. We begin by properly defining a target corpus of literature and drawing a sample set of documents designed to maximize information density. The corpus is then subjected to a statistical analysis to uncover high-frequency topic-specific terms and phrases as well as recurring text and patterns that are amenable to parsing with regular expressions. The statistical analysis also serves as the basis for our initial data model and a definition of the subject domain(s) encompassed by the corpus. The results are reviewed by subject matter experts and data curators, extracted terms are flagged for relevancy, grouped into appropriate topical categories, and working definitions are developed for those that are deemed relevant. These results are then used to refine topical data models, which evolve into proper relational database schemas, XML schemas and ontologies in a relatively short time frame. Term sets are mapped into our N4L Data Model, which accommodates rearrangements of taxonomies and refinement of the contextual and temporal meaning of terms. As terms are added, DOIs are assigned and made available for use in NamesforLife semantic annotation, tagging and indexing services and for incorporation into our ontologies.

While text mining, natural language processing and machine reasoning are all thought of as computational problems, our experience teaches that the human element, provided by subject matter experts and data curators is crucial if one is to obtain useable and meaningful results. Subject language terminologies (SLTs) are dynamic and may contain terms that have many nuanced meanings. SLTs often contain rare terms that significantly alter meaning but are not easily uncovered by machine learning methods. The use of the right tools at the right time is

important. Much of the early work of curators can be easily bootstrapped using familiar off-the-shelf tools, such as spreadsheets, for gathering preliminary data, manipulating the output of statistical analyses and quickly visualizing results. We have found that custom application development can be postponed until after databases are initially loaded and integrated into the NamesforLife environment. The other critical point that is often overlooked in text mining applications is the significant effect that changes in publishing technology can have on source documents, especially when older literature is included in the corpus. Minor differences in white space, formatting, character sets and typographical errors can have significant impact on precision, accuracy and recall.

### **Technology needs**

Many of the core technologies and resources needed to successfully mine metagenomic literature is either already in hand or is actively being developed for other applications in both the public and private sectors. The most efficient use of resources is to adapt proven computational methods to solve current problems rather than attempting to reinvent existing tools. On the other hand, expertise in the development of language resources needed to drive text mining applications in a meaningful way are in much shorter supply. Considerable effort is needed to develop normalized subject language terminologies based on actual usage by domain experts, to provide a mechanism for sustainably delivering this information, and to persistently integrate this knowledge into the literature. Failure to address this problem in a meaningful way will significantly diminish the potential return of any text mining initiative.

### **Candidate challenge problems**

- (1) To define a test corpus of the metagenomic literature based on the total published output in the field to date.
- (2) To define the relevant data types (including overlap with other related fields).
- (3) To develop a baseline terminology for each of those data types, along with domain values that are reported in the literature.

**Folker Meyer**  
Computational Biologist  
Institute for Genomics and Systems Biology  
Argonne National Laboratory  
9700 S. Cass Avenue, Lemont, IL  
U.S.A.

**Research interests in connection to metagenomics and text mining**

I am interested in the computational analysis of large metagenomic data sets. I am also the PI for the MG-RAST project.

**Priorities for text mining to improve data capture for metagenomics research**

Annotations for clonal genomes need to be improved and need to be migrated to a controlled namespace (e.g. SEED Subsystems).

**Technology needs**

Mostly it will be feasible by better CVs and better tools for using CVs. Minimal information checklists exist for metagenomic data but are missing for analysis.

## **Tatiana Tatusova**

Senior Scientist

National Center for Biotechnology Information

National Library of Medicine

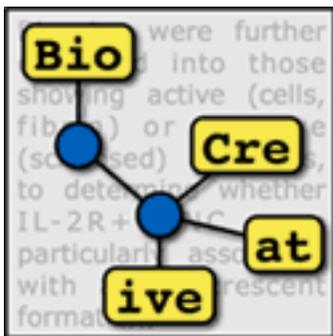
National Institutes of Health

45 Center Drive, Bethesda, Maryland 20814

Phone: 301-435-5756

Email: [Tatiana@ncbi.nlm.nih.gov](mailto:Tatiana@ncbi.nlm.nih.gov)

NCBI is serving the research community by maintaining the central repository for whole genome and metagenomics sequence data, as well as developing tools to facilitate archiving, search and analysis of these data. We have also been an active participant in the Genomic Standards Consortium, since NCBI has a strong interest in the development of minimum information standards and consistent capture of metadata for genomic and metagenomics sequences. A few years ago, we participated in the development of Habitat-Lite, a small set of structured terms to support capture of high-level information about the environment. Also NCBI is working on the Submission Portal – a centralized interface that will allow collecting multiple data types including sequence data and metadata in a single submission. We are eager to see the development of further structured vocabularies and tools, including text mining tools that can support reliable capture of metadata from the literature and potentially lead to the development of shared data model.



## POSTER ABSTRACTS

# Towards automatic large-scale curation of genomic variation: improving coverage based on supplementary material

Antonio Jimeno and Karin Verspoor

National ICT Australia and The University of Melbourne

A major thrust of modern biological research is the understanding of how genomic variation relates to disease. This information can be used for disease diagnosis, and increasingly, in the context of personalized medicine, to enable identification of effective disease treatments. There are large-scale efforts to catalogue the results of this research in structured databases, including in the OMIM database and the Human Gene Mutation Database. Much of this information is available only from unstructured sources, including the scientific literature. As such, there have been several systems developed to target extraction of mutations and other genetic variation from the literature. The performance of these tools has typically been evaluated intrinsically, that is, with respect to a gold standard set of annotations over a corpus of documents. Depending on the precise specification of the task, the gold standard corpus, and the tool tested, the performance of these tools have been claimed to achieve as high as perfect accuracy.

We instead perform an extrinsic evaluation of a mutation extraction tool. Specifically, we assess a mutation extraction tool with respect to the task of curation of the literature for the purpose of populating a database of genetic variation information. This is made possible due to the existence of several curated databases that catalogue genetic variants, and also provide links to the source literature supporting the variation and its disease association, including COSMIC and InSiGHT. Our analysis shows that the ability of the text mining tool to recover the mutations catalogued in the databases is far less than what would be expected based on the excellent performance on intrinsic evaluation. This effect has been previously observed; lack of access to the full text literature was suggested as a major contributor to the problem.

We show that the effect persists even when the full text article that was indicated to be the direct source of a mutation in a curated resource is available for processing. This can be argued to be the strongest scenario for high-recall text mining, where there exists an explicit link to the literature for a given piece of information, and access to the complete published narrative. In our evaluations we find that less than 3% of curated genetic variants are identified by text mining for COSMIC while just over 8% of variants in InSiGHT are recovered, even when full text is considered. We have explored several possible explanations for these results, including difficulties in linking genetic variants to specific genes, and the inclusion of data from high-throughput experiments. The coverage of variant annotation on COSMIC citations that are not

identified as high-throughput is improved but remains low. Relaxing the requirement of associating a variant to a specific gene improves recall to around 30%. We identify supplementary material as a primary source of relevant information. This result highlights the importance of processing all of the data associated with a publication, as well as identifying new research tasks around processing of semi-structured and inconsistently structured data resources related to publications.

**Keywords:** text mining; biocuration; genomic variation; mutation.

# WBI resources and participation in BioCreative IV

Torsten Huber<sup>1</sup>, Mariana Neves<sup>1</sup>, Tim Rocktäsc<sup>2</sup>, Philippe Thomas<sup>1</sup>, Michael Weidlich<sup>1</sup>, and Ulf Leser<sup>1</sup>

<sup>1</sup>Institute for Computer Science, Humboldt-Universität zu Berlin, Germany

<sup>2</sup>Department of Computer Science, University College London, UK

We present the contributions of the Knowledge Management in Bioinformatics (WBI) group to the BioCreative IV challenge. We participated in three of the five tasks: Track 1 – Interoperability BioC, Track 2 – Chemical and Drug Named Entity Recognition (CHEMDNER), and Track 5 – User Interactive Curation (IAT).

The Interoperability BioC task promoted the reuse of text mining modules by proposing the BioC XML format [1] as new standard for the biomedical natural language processing community. Participating teams were requested to provide a software module or other resource compliant with the BioC format. We contributed to this task in collaboration with the NICTA Victoria Research Lab and the Department of Computing and Information Systems in Melbourne, Australia, who developed a Java library [2] to convert corpora in the Brat stand-off format [3] to the BioC XML format. At WBI, we used this library to convert many of the corpora available in our BioNLP corpus repository [4], which provides syntax highlighting for more than 20 popular collections. Currently, 18 of these corpora are available to download in the BioC format.

The BioCreative IV CHEMDNER Task provides participants with the opportunity to compare methods for chemical named-entity recognition and indexing in a controlled environment. We contributed to this task with a novel method building on our existing chemical NER system ChemSpot [5]. Here, we used ChemSpots output and other novel general and domain-specific features, including the output of Oscar [6], as features for learning a linear-chain conditional random field based system. This system achieved a F1-measure of almost 82% on the development dataset.

The User Interactive task (IAT) aims to bring together biocurators and developers of text mining solutions. Teams had to submit a Web-based system for a biocuration task of their choice. In this task, WBI participated with a text mining pipeline previously developed in the scope of the CellFinder project [7], proposing to use the pipeline for curating gene/protein expression events in cells, tissues, and organs. Four curators were recruited for evaluation and suggested three biological topics of interest: kidney-related diseases in rat, human dendritic cells, and human

mesenchymal stem cells. Each curator validated gene/protein expression events automatically extracted from 30 Medline abstracts. A total of 634 expression events were obtained from the three datasets and approximately 35% of them (216 events) were validated as being correct.

**Keywords:** text mining; bioinformatics; named-entity recognition; data curation.

## **Resources**

[1] <http://bioc.sourceforge.net/>

[2] [https://bitbucket.org/nicta\\_biomed/brat2bioc](https://bitbucket.org/nicta_biomed/brat2bioc)

[3] <http://brat.nlplab.org/>

[4] <http://corpora.informatik.hu-berlin.de/>

[5] <https://www.informatik.hu-berlin.de/forschung/gebiete/wbi/resources/chemspot/chemspot/>

[6] <https://bitbucket.org/wwmm/oscar4/wiki/Home>

[7] <http://cellfinder.de/>

# Co-occurrence Interaction Nexus with Named-entity Recognition

Yi-Yu Hsu and Hung-Yu Kao

National Cheng Kung University, Taiwan

Co-occurrence Interaction Nexus with Named-entity Recognition (CoINNER) is a web-based tool that allows users to identify gene, chemical, disease, and action term mentions in the Comparative Toxicogenomic Database (CTD). As shown in our past study, different co-occurrence interactions can help curators greatly from manual curation. To further discover the interactions, CoINNER uses multiple advanced algorithms to recognize the mentions in the BioCreative 2013 CTD Track. CoINNER is developed based on a prototype system that annotated gene, chemical, and disease mentions in PubMed abstracts at BioCreative 2012 Track I (literature triage). We extended our previous system in developing CoINNER. The pre-tagging results of CoINNER were developed based on the state-of-the-art named entity recognition tools in BioCreative III. Next, a method based on the conditional random fields (CRFs) is proposed to predict chemical and disease mentions in the articles. Finally, action term mentions were collected by latent Dirichlet allocation (LDA). After taking advantage of aforementioned NER programs, CoINNER allows users to retrieve gene, chemical, disease, and action term mentions via a HTTP web request.

**Keywords:** Named-entity Recognition; Conditional random fields; Latent Dirichlet allocation.

## Resources

<http://140.116.245.192:8080/coinner/gene>

<http://140.116.245.192:8080/coinner/chemical>

<http://140.116.245.192:8080/coinner/disease>

[http://140.116.245.192:8080/coinner/action\\_term](http://140.116.245.192:8080/coinner/action_term)

# tmVar: A new machine learning method for mutation extraction in biomedical text

Chih-Hsuan Wei<sup>1</sup>, Bethany Harris<sup>2</sup>, Hung-Yu Kao<sup>3</sup>, and Zhiyong Lu<sup>1</sup>

<sup>1</sup> National Center of Biotechnology Information (NCBI), USA

<sup>2</sup> University of California, Irvine, USA

<sup>3</sup> National Cheng Kung University, Taiwan

Text-mining mutation information from the literature becomes a critical part of the bioinformatics approach for the analysis and interpretation of sequence variations in complex diseases in the post-genomic era. It has also been used for assisting the creation of disease-related mutation databases. Most of existing approaches are rule-based and focus on limited types of sequence variations such as protein point mutations. Thus, extending their extraction scope requires significant manual efforts in examining new instances and developing corresponding rules. As such, new automatic approaches are greatly needed for extracting different kinds of mutations with high accuracy.

Here we report tmVar, a text-mining approach based on conditional random field (CRF) for extracting a wide range of sequence variants described at protein, DNA, and RNA levels according to a standard nomenclature developed by the Human Genome Variation Society (HGVS). By doing so, we cover several important types of mutations that were not considered in past studies, and identify the informal mutation mentions (i.e., those that do not conform to any standard mutation nomenclature guidelines). We developed a CRF model to identify the mutation mentions in biomedical text and the assembled components (i.e., wild type, sequence position, and mutant) of the mentions. Using a novel CRF label model and feature set, our method achieves higher performance than a state-of-the-art method on both our corpus (92.2% vs. 78.1% in F-measure) and their own gold standard (93.9% vs. 89.4% in F-measure). These results suggest that tmVar is a high-performance method for mutation extraction from biomedical literature.

tmVar software and its corpus of 500 manually curated abstracts are publicly available at: <http://www.ncbi.nlm.nih.gov/CBBresearch/Lu/downloads/tmVar/>. Results on PubMed articles are available in PubTator: <http://www.ncbi.nlm.nih.gov/CBBresearch/Lu/Demo/PubTator>.

**Keywords:** Biomedical text mining; Mutation recognition; Conditional random field.

## **Acknowledgments**

This research was supported by the NIH Intramural Research Program, National Library of Medicine. BH was supported in part by an appointment to the NLM Associate Fellowship Program sponsored by the National Library of Medicine and administered by the Oak Ridge Institute for Science and Education.

## **References**

1. Chih-Hsuan Wei, Bethany R. Harris, Hung-Yu Kao, Zhiyong Lu (2013) "tmVar: A text mining approach for extracting sequence variants in biomedical literature" *Bioinformatics*, 29 (11), 1433-1439, doi:10.1093/bioinformatics/btt156

# DNorm: A New Method and Tool for Disease Name Normalization

Robert Leaman, Rezarta Islamaj Dogan, Ritu Khare, Chih-Hsuan Wei, and Zhiyong Lu

National Center of Biotechnology Information (NCBI)

Disease is a central topic in biomedical research, and determining which diseases are referenced by a text – the task of disease name normalization or grounding – is important to many lines of clinical and biological inquiry. Automated methods for disease name normalization may be useful for reducing the human effort required for indexing abstracts (e.g. MeSH), biocuration projects (e.g. the Comprehensive Toxicogenomics Database, CTD), and reviewing clinical records (e.g. for epidemiological studies or clinical trial prescreening).

We have created DNorm, an open-source system for disease name normalization. DNorm employs conditional random fields to locate disease mentions (named entity recognition) [1]. Each disease mention found is then normalized using a novel machine learning methodology based on pairwise learning to rank [2]. This new methodology represents a high-performing and mathematically principled framework for learning similarities between mentions and concept names directly from training data, and is capable of learning synonymy, polysemy, and relationships that are not 1-to-1.

DNorm has been evaluated on two corpora. The NCBI Disease Corpus consists of PubMed abstracts and is annotated with disease concepts from MeSH and OMIM [3]. We compared the results of DNorm to several existing techniques including a lexical lookup (with string normalization via Norm), MetaMap, Lucene and cosine similarity. DNorm achieves 0.782 micro-averaged F-measure and 0.809 macro-averaged F-measure, a significant increase of 0.121 and 0.098, respectively, over the highest performing baseline method [4]. We also evaluated DNorm on clinical notes by participating in the ShARe / CLEF eHealth Task 1b, where disease mentions are normalized to SNOMED-CT [5]. DNorm achieved an accuracy of 0.589, the highest performance among the 17 participating teams by a margin of 0.046 [6].

The source code for DNorm is available at <http://www.ncbi.nlm.nih.gov/CBBresearch/Lu/Demo/DNorm>, along with a web-based demonstration and links to the NCBI disease corpus. Results on PubMed abstracts are available in PubTator: <http://www.ncbi.nlm.nih.gov/CBBresearch/Lu/Demo/PubTator>.

**Keywords:** Disease normalization; Machine learning; Pairwise learning to rank

# Mining the Impact of Phosphorylation on PPI in Full Length Scientific Articles

Catalina O. Tudor<sup>1,2</sup>, Cecilia N. Arighi<sup>1,2</sup>, Cathy H. Wu<sup>1,2</sup>, K. Vijay-Shanker<sup>1</sup>

<sup>1</sup> Department of Computer and Information Sciences, University of Delaware, USA

<sup>2</sup> Center of Bioinformatics and Computational Biology (CBCB), University of Delaware, USA

Post-translational modifications play a fundamental role in regulating the activity, location and function of a wide range of proteins. In particular, protein phosphorylation by protein kinases and de-phosphorylation by phosphatases play a major role in almost all critical cellular events, such as cell metabolism regulation, cell division, cell growth and differentiation. Often, protein phosphorylation results in some functional impact. For instance, proteins can be phosphorylated on different residues, leading to either activation or down-regulation of their activities, alternative subcellular locations and/or interaction with distinct binding partners. Extracting this type of information from scientific literature is critical for connecting phosphorylated proteins with kinases and interaction partners, along with their functional outcomes, for knowledge discovery from phosphorylation protein networks.

We have developed the eFIP (Extracting Functional Impact of Phosphorylation) text mining system, which combines several natural language processing techniques to find relevant *abstracts* mentioning phosphorylation of a given protein together with indications of protein–protein interactions (PPIs) and potential evidences for impact of phosphorylation on the PPIs [1]. As a participant in the BioCreative-2012 Interactive Text Mining track, the performance of eFIP was evaluated on document retrieval ( $F$ -measures of 78–100%), sentence-level information extraction ( $F$ -measures of 70–80%) and document ranking (normalized discounted cumulative gain measures of 93–100% and mean average precision of 0.86). The utility and usability of the eFIP web interface were also evaluated during the BioCreative Workshop. The use of the eFIP interface provided a significant speed-up (~2.5-fold) for time to completion of the curation task. Additionally, eFIP significantly simplifies the task of finding relevant articles on PPI involving phosphorylated forms of a given protein.

We are currently extending the eFIP system to include *full length scientific articles*. Based on a preliminary study, in which we measured the coverage of functional impact of phosphorylation on protein-protein interaction, we observed that over 92.6% of scientific articles mention useful information only in sections other than the abstract. This is a significant number when compared to 3.3% of scientific articles mentioning useful information only in the abstract section. The rest of 4% are articles that contain useful information in both the abstract section and at least one

other section. However, in most of these cases, we were able to find additional useful information in the rest of the article that was not mentioned in the abstract. Primarily, the Results and Discussion sections contain the most information that is relevant to the task of mining impact of phosphorylation on protein-protein interaction. These are followed by the Introduction, Background, Materials and Methods, and Conclusion sections, in this order.

**Keywords:** biomedical text mining, relation extraction

## **Funding**

This work was supported by National Science Foundation grant ABI-1062520 and National Institutes of Health grants 5G08LM010720-02 and 5R01GM080646-06.

## **References**

1. Catalina O Tudor, Cecilia N Arighi, Qinghua Wang, Cathy H Wu, K Vijay-Shanker. The eFIP system for text mining of protein interaction networks of phosphorylated proteins, Database 2012.