Performance and interoperability assessment of Disease Extract Annotation Server (DEAS)

Jitendra Jonnagaddala*[1,2], Hong-Jie Dai[3,4],
Chen-Kai Wang[5], Po-Ting Lai[6]

[1] School of Public Health and Community Medicine, UNSW Sydney, Australia
[2] Prince of Wales Clinical School, UNSW Sydney, Australia
[3] Department of Computer Science and Information Engineering, National Taitung University,
Taitung, Taiwan, R.O.C,
[4]Interdisciplinary Program of Green and Information Technology, National Taitung University,
Taitung, Taiwan, R.O.C,
[5]Graduate Institute of Biomedical Informatics, Taipei Medical University,
Taipei, Taiwan, R.O.C,
[6]Department of Computer Science, National Tsing-Hua University,
Hsinchu, Taiwan, R.O.C.

[*1,2]z3339253@unsw.edu.au; [3,4]hjdai@nttu.edu.tw;
[5]dennisckwang@gmail.com; [6]potinglai@gamil.com

Abstract. Over the past decade, many biomedical information extraction tools have been developed. Some of these tools are provided with access via web services. However, many of these tools are not interoperable with web services performance not evaluated. In this study, we implemented and evaluated the performance of an interoperable web service that annotates disease entities called Disease Extract Annotation Server (DEAS). The DEAS evaluation was carried out over a period of two months. Interoperability, stability, speed and batch processing capabilities of DEAS were evaluated. DEAS was able to process documents from multi data sources and types with response times varying from one second to six seconds per document. The performance evaluation assisted in improving the underlying CRF-based disease entity recognition pipeline. In future, we would like to support more data format standards and also improve DEAS performance by employing distributed and parallel processing techniques.

Keywords. Named entity recognition; biomedical entity recognition; biomedical meta-services; disease annotation server

*Corresponding author

1 Introduction

The exponential increase of published biomedical literature is creating a great demand to effectively retrieve and extract relevant information. Many methods have been proposed to extract unstructured information effectively. For example, Dai. et al presented methods to normalize species and gene/protein mentions [1]. However, often these methods focus on a very specific extraction or retrieval tasks. Also, sometimes these methods are not interoperable. The data formats are different and are usually developed and supported for different platforms. It is also difficult to unify extracted information from multiple information extraction systems. To some extent this can be resolved by implementing meta-services [2]. Meta-services integrate multiple information extraction (IE) and information retrieval (IR) systems by enforcing standards using web services where suitable. Web services are software packages designed to support communications by wide range of devices and platforms using web standards. BioC, a data interchange format is a good example which can be adapted into meta-services [3].

The Biomedical Annotation meta-server[2] (BeCalm) platform supports meta-services by providing access to various annotation servers. BeCalm reinforces a minimal set of standards to harmonize various annotation servers; which in turn can be subjected to comparative assessment and continuous evaluation. In this study, we implemented an interoperable REST (Representational State Transfer) based web services that annotate disease entities based on the BeCalm meta-server standards. We call our implementation as Disease Extract Annotation Server (DEAS). We also evaluated the performance of the DEAS by carrying out an evaluation which lasted for two months as part of the BioCreative V.5 Technical interoperability and performance of annotation servers (TIPS) task[3].

2 Methods

The DEAS[4] is an extension to our previous work [4, 5]. In specific for the purpose of this study, we extended our previous web services to support BeCalm meta-server API specifications. The DEAS mainly include

[2] http://www.becalm.eu/api/

[3] http://www.becalm.eu/pages/biocreative

[4] https://github.com/TCRNBioinformatics/DiseaseExtract

three components: 1) data retrieval 2) web services and 3) infrastructure layer. DEAS supports automatic retrieval of biomedical articles, patents and abstracts from three different data sources (table 1) in the data retrieval layer. In this component, the DEAS retrieves the documents requested by a meta-sever or another other client and passes it over to the web services component for further handling.

Data Source	Data type	Data source details
PMC and PubMED	Biomedical articles	https://eutils.ncbi.nlm.nih.gov/entrez/eutils/
Patent Server	Patents	http://193.147.85.10:8087/patentserver/json/
Abstract Server	Biomedical Abstracts	http://193.147.85.10:8088/abstractserver/json/

Table 1: DEAS Data retrieval sources

DEAS web services were developed using the REST paradigm [6]. REST is an architectural style that allows to create communication services using The Hypertext Transfer Protocol (HTTP) standard. REST is most commonly known for its scalability and light-weight characteristics. We have used Swagger[5] framework to develop, document, and consume the DEAS web services. The DEAS currently supports both requests and responses in JSON format. However, interoperable BioC format [3] is not supported, although it is supported in the underlying disease entity recognition system. A sample DEAS web services request and response is presented in table 2. In this sample, DEAS automatically retrieves a patent from the patent server and annotates the content of the patent for disease entities. Please refer to the DEAS documentation for complete list of web services supported.

```
    BeCalm Request:
    getAnnotations:
    {
      "name":"BeCalm",
      "method":"getAnnotations",
      "becalm_key":"0000000000",
      "custom_parameters" :{,
      "parameters" : {
         "documents":[{"document_id":"CA2073855C","source":"PATENT
 SERVER"}],
```

[5] http://swagger.io/

```
        "types": ["Disease"],
        "communication_id":1000
    }
}
```

DEAS Response:

```
{"status":200,"success":true,"key":"0000000000"}
```

Table 2: DEAS API Example

The data retrieval and web services components were wrapped around an infrastructure layer where we used a Linux machine with 1GB memory and 1 Intel Xeon 2.4 GHz CPU. The stability, speed and batch processing capabilities of DEAS were evaluated using the metrics, constructed mainly of the number of predictions made and response times. The response times here refer to the time taken by an annotation server such as DEAS to respond to the request made by a client, such as BeCalm. If the request includes processing multiple documents, the response time is expected to be higher. Similarly, the response time could vary depending on the infrastructure and other underlying components of an annotation server.

3 Results and Discussion

The DEAS evaluation began on 1st of February and ended on 31st of March 2017. Table 3 presents the performance of our disease annotation server developed by each data source. DEAS was able to process majority of the requests from BeCalm server where either patent server or, PMC and PubMed server is the data source. However, the majority of the requests from the abstract server were not processed which is reflected in terms of the number of exceptions and predictions. It is also important to note that the large maximum response times are mainly due to the exceptions caused.

	Patent server	Abstract server	PMC and PubMed
Total processing time	26 days 21h:10m 8: s	3 days 14h:0m 1: s	3 days 02h:5m 0: s
Total predictions	189 k	0 k	374 k
Mean predictions per request	3.3	0	4.9
Exceptions/Total requests	7177/82213	23798/135938	7920/92811
Minimum response time (seconds)	1.72	1.65	1.84

Maximum response time (seconds)	30249.38	895.45	24568.9
Mean response time (seconds)	3116.81	2.78	967.52

Table 3: Performance by data source

Figure 2 presents the DEAS daily response time from 15th of March to 31st of March. The response times were collected at irregular intervals during the day and the figure illustrates that our response rates varied somewhere between 2 seconds and 12 seconds.

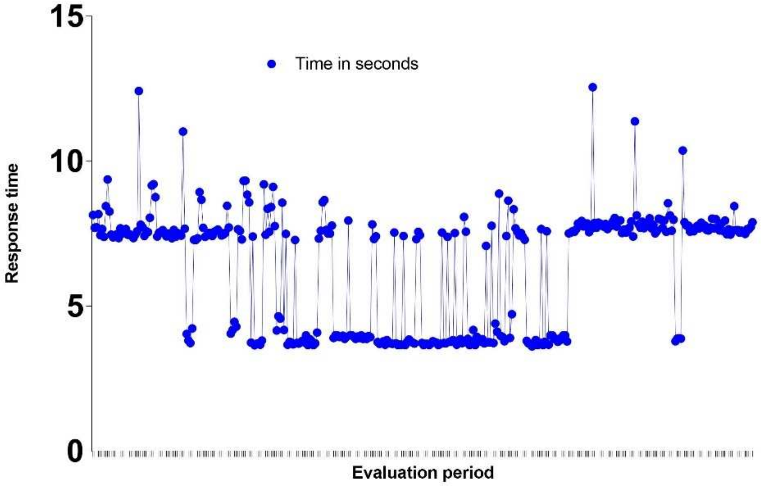

Figure 2: Historical server response timeout

The batch processing capabilities were evaluated by posting requests with multiple documents in each request varying from 10 to 2400 documents per request. Table 4 presents the metrics for batch processing. The DEAS capability to process in bulk has been significantly improved by reducing the processing time from 5.9 seconds per document (40 documents) to 1.19seconds per document (2400 documents).

Total Time(mm:ss.ms)[6]	Time(s)/documents	#Documents	#Predictions
47:38.9	1.19122	2400	5667
15:03.0	0.75254	1200	2546
06:21.0	0.76208	500	1031
03:30.1	2.2116	95	262

[6] minutes: seconds. milliseconds

03:54.4	5.86047	40	104
00:46.9	4.69455	10	13

Table 4: Batch processing performance metrics

The initial results from this study assisted us in improving our underlying core entity recognition pipeline which was based on conditional random fields (CRFs) [7]. Our core pipeline[4] used Stanford PTBTokenizer[7] and was failing to process documents containing non-UTF characters. In other words, DEAS could not find any disease annotations and was sending empty responses to the BeCalm meta-server. This assisted us in identifying and fixing the underlying issue with the tokenizer. Similarly, we were able to detect and fix memory related performance issues resulting to improved scalability and reliability of the underlying pipeline. On the other hand, we believe that the BeCalm meta-server can also be improved in the API, documentation and communication areas. For example, at the beginning of the evaluation period BeCalm did not require an annotation server to support the Abstract server. However, the addition of the Abstract server data source requirement was not communicated. Thus, our annotation server could not annotate majority of the requests with the Abstract server as the data source (Table 3). One major limitation of this study is the performance metrics used. The performance is evaluated for speed, stability and batch processing capabilities only, but not for the quality or accuracy of underlying disease entity recognition which typically is reported using precision, recall and F-measure.

4 Conclusion

In conclusion, we present the performance and interoperability assessment of DEAS, a disease extract annotation server. DEAS employs CRF-based entity recognition to extract disease entities from biomedical articles and patents. The evaluation results show that DEAS can effectively interoperate with the BeCalm meta-server and process documents from multiple sources. In our future work, we also would like to support the processing of documents with non-UTF characters. Additionally, aside from supporting requests and responses in JSON format, we would also like to enable support for BioC format. Lastly, we would also like to

[7] https://nlp.stanford.edu/software/tokenizer.shtml

decrease response times by employing distributed and parallel processing techniques.

5 Acknowledgment

This study was conducted as part of the electronic Practice Based Research Network (ePBRN) and Translational Cancer research network (TCRN) research programs. We would like to thank the organizers of BioCreative V.5 TIPS Track for providing us, with the sample code to develop BeCalm compliant APIs.

REFERENCES

1. Dai, H.-J., et al., *NTTMUNSW BioC modules for recognizing and normalizing species and gene/protein mentions.* Database, 2016. **2016**: p. baw111-baw111.
2. Leitner, F., et al., *Introducing meta-services for biomedical information extraction.* Genome Biology, 2008. **9**(Suppl 2): p. S6-S6.
3. Comeau, D.C., et al., *BioC: a minimalist approach to interoperability for biomedical text processing.* Database, 2013. **2013**: p. bat064-bat064.
4. Jonnagaddala, J., et al., *Improving the dictionary lookup approach for disease normalization using enhanced dictionary and query expansion.* Database, 2016. **2016**: p. baw112-baw112.
5. Jonnagaddala, J., et al. *Recognition and normalization of disease mentions in PubMed abstracts*. in *Proceedings of the fifth BioCreative challenge evaluation workshop, Sevilla, Spain, September 9-11, 2015*. 2015.
6. Fielding, R.T., *Architectural styles and the design of network-based software architectures*. 2000, University of California, Irvine. p. 162.
7. Sutton, C. and A. McCallum, *An introduction to conditional random fields.* Foundations and Trends® in Machine Learning, 2012. **4**(4): p. 267-373.