Protein-protein Interaction Passage Extraction Using the Interaction Pattern Kernel Approach for the BioCreative 2015 BioC Track

Yung-Chun Chang[1,2], Yu-Chen Su[3], Chun-Han Chu[1], Chien Chin Chen[2] and Wen-Lian Hsu[1]

[1]Institute of Information Science, Academia Sinica, Taipei, Taiwan
[2]Department of Information Management, National Taiwan University, Taipei, Taiwan
[3] Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan

[1]{changyc, johannchu, hsu}@iis.sinica.edu.tw;
[2]patonchen@ntu.edu.tw;
[3]s103062704@m103.nthu.edu.tw;

Abstract. Discovering the interactions between proteins mentioned in biomedical literatures is one of the core topics of text mining in the field of life science. In this paper, we propose a system under interaction pattern generation approach to capture frequent PPI patterns in text with the use of official BioC API and Semantic Class Labeling. We also present an interaction pattern tree kernel method that integrates the PPI pattern with convolution tree kernel to extract protein-protein interactions. Empirical evaluations on the LLL, IEPA, and HPRD50 corpora demonstrate that our method is effective and outperforms several well-known PPI extraction methods.

Keywords. Text Mining; Protein-Protein Interaction; Interaction Pattern Generation; Interaction Pattern Tree Kernel

1 Introduction

With the growing number of research papers, researchers now have difficulty in retrieving those that exactly fulfill their needs. As for life scientists, relationships between entities mentioned in these papers are the major target of interest. Among biomed relation types, protein–protein interaction (PPI) extraction is becoming critical in the field of molecular biology due to demands for automatic discovery of molecular pathways and interactions in the literature.

Most PPI extraction methods can be treated as supervised learning, in which feature-based and kernel-based approaches are frequently used. However, feature-based methods often have difficulty finding effective features to extract entity relations. To solve this problem, kernel-based methods have been proposed to explore features in a high dimensional space by employing a kernel to calculate the similarity between two objects. Erkan et al. [1] defined two kernel functions based on cosine similarity and the edit distance among the shortest paths between protein names in a dependency parse tree. Satre et al. [2] developed the Akane PPI that extracts features using the combination of a deep syntactic parser to capture the semantic meaning of sentences with a shallow dependency parser for tree kernels. Moschitti et al. [3] adopted a partial tree kernel (PT) which is more flexible by virtually allowing any tree sub-structures without major constraints.

For the extraction of PPIs, we propose an interaction pattern generation approach to capture frequent PPI patterns. Furthermore, to identify interactions between proteins, we developed an interaction pattern tree kernel that integrates the shortest path-enclosed tree (SPT) structure with generated PPI patterns to support vector machines (SVM). The results of experiments demonstrate that the interactive pattern tree kernel method is effective in extracting PPI. Also, the proposed pattern generation approach successfully exploits the interaction semantics of text by capturing frequent PPI patterns. Consequently, the method outperforms the tree kernel-based PPI method [3], the feature-based PPI method [4], and the SPT detection method [5], which is widely used to identify relations between named entities.

2 System Architecture

Figure 1 shows the proposed interaction extraction method that is comprised of two key components: *interaction pattern generation* and *interaction pattern tree construction*. The interaction pattern generation component aims to automatically generate representative patterns of mentioned interactions between proteins. Then, the interaction pattern tree construction integrates the syntactic and content information with generated interaction patterns for representation of text. Finally, the convolution tree kernel measures similarity between interaction pattern tree structures for SVM to classify interactive expressions.

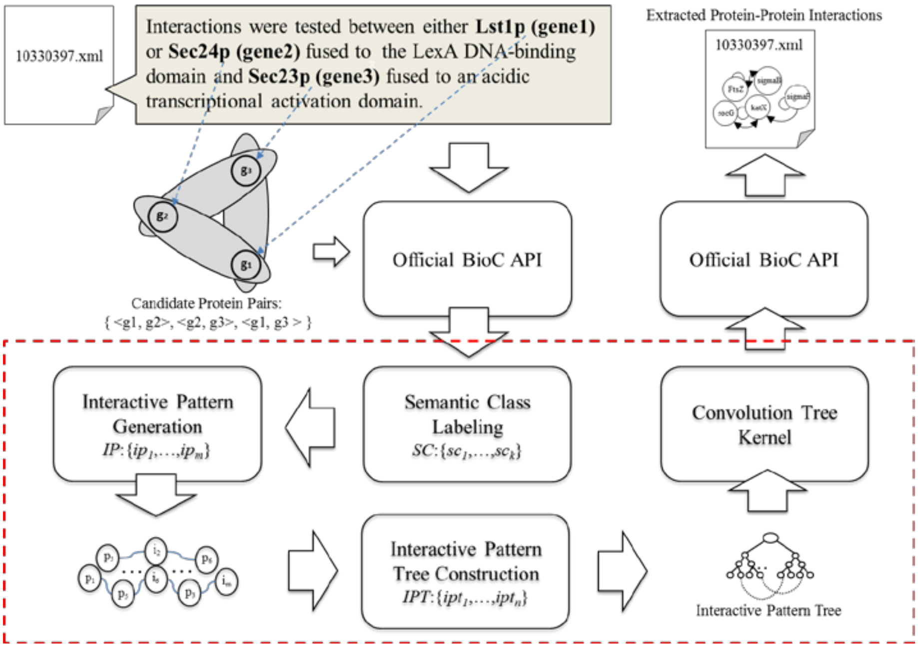

Fig. 1. The interaction extraction method

To capture possible PPI candidates, our goal is to first retrieve every sentence in an article that has at least two kinds of protein names. This is done through a customized function. With the processed sentences as input to our main system, we convert them into candidate units via normalization and parsing. Consider sentence s_1: "*Interactions were tested between either Lst1p or Sec24p fused to the LexA DNA-binding domain and Sec23p fused to an acidic transcriptional activation domain*" as an example. It contains recognized genes "*Lst1p*", "*Sec24p*" and "*Sec23p*" which are labeled as $\{g_1, g_2, g_3\}$, respectively. We process each sentence with different pairs of genes that corresponding normalized and parsed sentences are added to form the expanded candidate units:$\{s_1, n_1, p_1, g_1, g_2\}$, $\{s_1, n_2, p_2, g_2, g_3\}$ and $\{s_1, n_3, p_3, g_1, g_3\}$. The instances then undergo semantic class labeling (SCL), where proteins would first be identified and tagged. To illustrate the process of semantic class labeling, consider the instance I_n = "*Abolition of the gp130 binding site in hLIF created antagonists of LIF action*", as shown in Fig. 2. First, "*gp130*" and "*hLIF*" are two given protein names, as tagged *PROTEIN1* and *PROTEIN2* respectively. Then, we stem remaining tokens by using porter stemming algorithm [6], followed by *trigger word labeling* with our word list extracted from a BioNLP corpus [7]. Evidently, SCL can group the synonyms together by the same label, enabling us to find distinctive and prominent semantic classes for PPI expression in the upcoming stage.

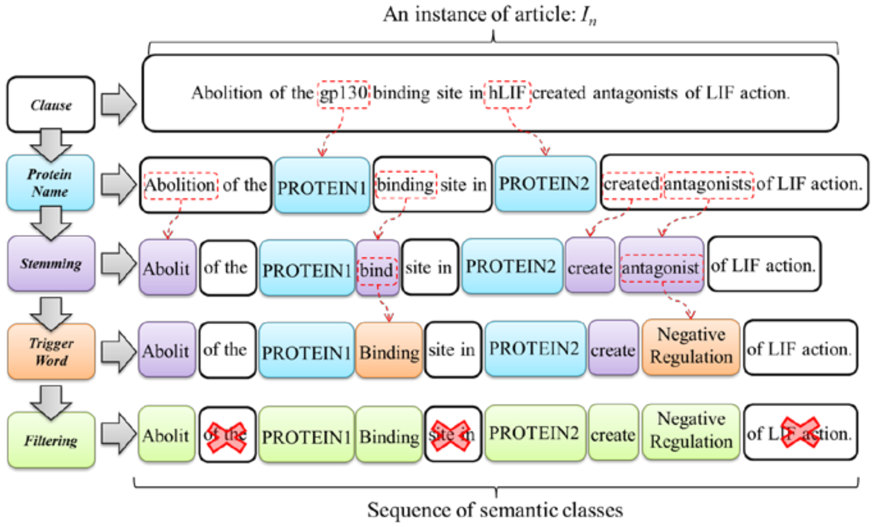

Fig. 2. Semantic class labeling process

After semantic classes were labeled, we construct a graph describing the strength of relations between these classes based on their co-occurrence. Since semantic classes are of an ordered nature, the graph is directed and can be made with association rules. In order to avoid the generation of frames with insufficient length, we empirically set the minimum support of a semantic class as 20 and minimum confidence as 0.5 in our association rules. After constructing all semantic graphs, we then generate semantic frames by applying the random walk theory [10] in search of high frequency and representative classes for each topic. Consider the generated interaction pattern " [*Positive_regulation*] -> [*Regulation*] -> [*Gene_expression*] -> [*PROTEIN1*]" as an example, an instance like "*In the final set of experiments, we explored the possible participation of CBP in Stat1 driven gene expression.*", which contains corresponding trigger words, can match all of the components of the semantic frame. Nevertheless, although the random walk process can help generate frames from frequent patterns in semantic graphs, it can also create redundancy. Therefore, a merging procedure is thus required to eliminate the redundant results by retaining patterns with long length and high coverage, while disposing of bi-gram patterns that are completely covered by another pattern. Moreover, the reduction of the semantic class space provided by pattern selection is critical. It allows the execution of more sophisticated text classification algorithms, which leads to improved results. These algorithms, however, cannot be executed on the original semantic class space due to their high execution time [11]. Hence, we only pick patterns closely associated with an interaction to improve the performance of PPI extraction. A PPI instance is later represented by the interaction pattern tree (IPT) structure, which is the enhancement of SPT.

Finally, the generated interaction patterns can be used to capture the most prominent and representative patterns for expressing PPI. Highlighting interaction patterns closely associated with PPIs in an IPT would improve the interaction extraction performance. For each IPT that matched an interaction pattern, we add an IP tag as a child of the tree root to incorporate the interactive semantics into the IPT structure. Using the trained SVM classifier, the IPTs are classified as either positive or negative regarding whether at least one interaction exists. We then utilize the official library again to add annotations for the corresponding positive instances, which are recognized as containing protein-protein interactions, to the provided NER dataset.

3 Results and Discussion

We evaluated our method with three publicly available corpora: LLL (P: 164, N: 166), IEPA (P: 335, N: 482), and HPRD50 (P: 163, N: 270) [8]. All corpora are parsed with the Stanford parser to generate the output of parse tree and part-of-speech tagging. In our implementation, we used Moschitti's tree kernel toolkit [3] to develop the convolution kernel of an IPT. We then performed a 10-fold cross validation [11] on all corpora. The F_1-measure [11] is used to determine the relative effectiveness of the compared methods for evaluation. We exploit the macro-averaged score to show the overall performance across three different corpora for each evaluation metric.

The performance of our system is compared with several PPI extraction methods. As shown in Table 1, the proposed method significantly outperforms SPT and AkanePPI. Furthermore, the syntax tree-based kernel methods (PT) only examined the syntactic structures of text and cannot sense the semantics of protein interactions. By contrast, our method analyzes both the semantics and content (i.e., PPI patterns) of text. It is worth noting that syntax tree-based kernel methods are often only on par with the co-occurrence approach in terms of F1-measure. On the relatively small LLL corpus, their results practically coincide with that of the co-occurrence approach. The rich-feature-based (RFB) and Cosine also outperformed the SPT, AkanePPI and syntax tree-based kernel methods as they incorporate dependency features to distinguish protein-protein interactions. However, while Cosine can accomplish higher performance by further considering term weighting, it has difficulty in representing word relations. Our method, on the other hand, can extract word semantics and generate PPI patterns that can capture long-distance relations among them, hence achieving a better result.

Table1. The interaction extraction performance of the compared methods

System	***LLL***	***IEPA***	***HPRD50***	***Macro-average***
	Precision, Recall, F1-measure (%)			
SPT	56.4 / 96.1 / 69.6	55.5 / 28.8 / 37.1	46.2 / 13.4 / 20.8	52.7 / 46.1 / 42.5
AkanePPI [2]	**76.7** / 40.2 / 52.8	**66.2** / 51.3 / 57.8	52.0 / 55.8 / 53.8	**65.0** / 49.1 / 54.8
PT [3]	56.2 / 97.3 / 69.3	63.1 / 66.3 / 63.8	54.9 / 56.7 / 52.4	58.1 / 73.4 / 61.8
RFB[4]	72.0 / 73.0 / 73.0	64.0 / 70.0 / 67.0	**60.0** / 51.0 / 55.0	65.3 / 64.7 / 65.0
Cosine [1]	70.2 / 81.7 / **73.8**	61.3 / 68.4 / 64.1	59.0 / 67.2 / 61.2	63.5 / 72.4 / 66.4
Our method	59.9 / 94.4 / 71.6	52.2 / 88.1 / **65.2**	59.3 / 83.0 / **67.3**	57.1 / 88.5 / **68.0**

4. Concluding Remarks

To this end, we have proposed an effective interaction pattern generation approach for acquiring PPI patterns. We have also developed a method that improves over the SPT structure by including in PPI patterns to analyze the syntactic, semantic, and content information in text. It then exploits the derived information to identify PPIs in biomedical literatures. Our results show that the proposed method outperforms several well-known ones.

Acknowledgment

This research was supported by the National Science Council of Taiwan under grant NSC102-3113-P-001-006, MOST103-2319-B-010-002 and MOST103-3111-Y-001-027.

REFERENCES

1. G. Erkan, A. Özgür, and D. R. Radev. Semi-supervised classification for extracting protein interaction sentences using dependency parsing. In *Proceedings of the 2007 Joint Conf. on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages228-237, 2007.
2. R. Satre, K. Sagae, and J. Tsujii. Syntactic features for protein-protein interaction extraction. In *Proceedings of the 2nd international symposium on languages in biology and medicine*, pages 6.1-6.14, 2007.
3. A. Moschitti. Efficient convolution kernels for dependency and constituent syntactic trees. In *Proceedings of the 17th European Conf. on Machine Learning*, pages 318-329, 2006.
4. S. Van Landeghem, Y. Saeys, B. De Baets, and Y. Van de Peer. Extracting protein-protein interactions from text using rich feature vectors and feature selection. In *Proceedings of 3rd International Symposium on Semantic Mining in Biomedicine*, pages 77-84, 2008.
5. M. Zhang, J. Zhang, J. Su and G.D. Zhou. A Composite Kernel to Extract Relations between Entities with both Flat and Structured Features. *COLINGACL*, pages 825-832, Sydney, Australia, 2006.
6. M. F. Porter. An algorithm for suffix stripping, in *Readings in Information Retrieval*, Karen Sparck Jones and Peter Willet (ed), San Francisco: Morgan Kaufmann, 1997.
7. J.D Kim, T. Ohta, S. Pyysalo, Y. Kano, and J. Tsujii. Overview of BioNLP'09 shared task on event extraction, In *Proceeding of the Workshop on Current Trends in Biomedical Natural Language Processing: Shared Task*, pages 1-9, 2009.
8. A. Airola, S. Pyysalo, J. Bjo¨rne, T. Pahikkala, and F. Ginter. All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning.*BMC Bioinformatics 9: S2*, 2008.
9. http://corpora.informatik.hu-berlin.de/
10. L. Lovász. Random walks on graphs: asurvey. Janos Bolyai Mathematical Society, Budapest 2, pages 1-46, 1993.
11. C.D. Manning and H.Schütze. Foundations of statistical natural language processing: MIT Press, Cambridge, Massachusetts, 1stedn., 1999.