A Hybrid Approach for Biomedical Event Extraction

Xuan Quang Pham
Faculty of Information Technology
University of Science
Ho Chi Minh City, Vietnam
pxquang@fit.hcmus.edu.vn

Minh Quang Le
Faculty of Information Technology
University of Science
Ho Chi Minh City, Vietnam
mquang88@gmail.com

Bao Quoc Ho
Faculty of Information Technology
University of Science
Ho Chi Minh City, Vietnam
hbquoc@fit.hcmus.edu.vn

Abstract

In this paper we propose a system which uses hybrid methods that combine both rule-based and machine learning (ML)-based approaches to solve GENIA Event Extraction of BioNLP Shared Task 2013. We apply UIMA[1] Framework to support coding. There are three main stages in model: Pre-processing, trigger detection and biomedical event detection. We use dictionary and support vector machine classifier to detect event triggers. Event detection is applied on syntactic patterns which are combined with features extracted for classification.

1 Introduction

The data in biomedicine is continuously bigger and bigger because of the incredible growth of literatures, researches or documents in that field. This huge resource has been attracted a significant interest on developing methods to automatically extract biological relations from text. Most of them are binary relation such as protein-protein interactions, gene-disease and drug-protein relations. However there are more complex events in origin biomedical data. The BioNLP Shared Task (BioNLP-ST) is one of the efforts to promote extracting fine-grained and complex relations in biomedical domain.

BioNLP Shared Task 2013 has the six event extraction tasks such as GENIA Event Extraction (GE), Cancer Genetics (CG), Pathway Curation (PC), Gene Regulation Ontology (GRO), Gene Regulation Network (GRN) and Bacteria Biotopes (BB). The GE task has three subtasks, task 1 is detection of events with their main arguments, task 2 extends this to detection of sites defining the exact molecular location of interactions, and task 3 adds the detection of whether events are stated in a negated or speculative context.

In event extraction, common approaches use Rule-based (Kaljurand et al., 2009; Kilicoglu and Bergler, 2011), Machine Learning (ML)-based (Björne at al., 2009; Miwa et al., 2010) and hybrid methods (Ahmed et al., 2009; Riedel, McClosky et al., 2011). Recently, (Riedel et al., 2011) present an approach based on optimization of scoring sets of binary variables. The model and a variant model (hybrid model) gained the second and first place in BioNLP-ST 2011, proving the effect of their approach. According to the summaries of BioNLP-ST 2009 and 2011 (Kim., 2011), the results of ML-based method are better than the rule-based method. However ML is nontrivial to apply. The summary also indicates that high precision, for simple events, can be achieved by Rule-based approach.

In this paper, we present our work for GE task. We try to apply our knowledge from general information extraction to a specific domain, biomedicine. We propose a system which uses hybrid methods that combine both rule-based and machine learning (ML)-based approaches.

2 Proposed approach

We use the UIMA framework to support all steps of the model. The UIMA is an open source framework for analyzing general unstructured data. This framework is applied mainly to save our time of coding. Thanks to it, we can take advantage of some developed modules and improve them easier. All modules are described in detail in the following sections.

2.1 Pre-processing

At first, we need to convert input texts into objects of the framework to store and process later.

[1] http://uima.apache.org/

Proceedings of the BioNLP Shared Task 2013 Workshop, pages 121–124,
Sofia, Bulgaria, August 9 2013. ©2013 Association for Computational Linguistics

From this part to the end, all analyzed and annotated results will be stored in those objects. Secondly, natural language processing (NLP) is applied. It includes splitting sentences, tokenized, POS tagger and deep parser. There are various libraries in NLP, both general and specific domain but we select the McClosky-Charniak-Johnson Parser[2] for syntactic analyses. That parser is improved from the Stanford parser with a self-trained biomedical model. According to the shared task's statistics (Kim et al., 2011), it is used by groups achieving high results. In addition, the NLP data of all datasets are prepard and provided for participants. We read and convert the given results into our framework to use in further processing. We also add other information on the token such as stems of single token (using the Snowball stemmer), id in the sentence and the nearest dependent/governor token.

Finally, we convert all the annotated proteins of input into UIMA. These proteins are candidate arguments for events. Similar to NLP data, the annotations are provided by the shared task as supporting resources. Each single file has a separate list of given proteins appearing in its content.

2.2 Trigger detection

In the shared task 2011, we used simple rules and dictionaries to annotate triggers or entities (Le, M.Q., 2011), but there were many ambiguities. Furthermore, a candidate trigger can belong to a few types. Consequently, the performance of that method was fairly poor. Thus, we decided to change to a machine learning approach, which needs less domain knowledge, in the shared task 2013.

We need to classify a token into one of eleven groups (nine for Event Trigger, one for Entity and one for nothing). We separate tokens instead of phrases for the following reasons. Firstly, Event Triggers and Entities which cover single token are more popular. Secondly, the official evaluation of the shared task is approximate span. The given span belonging to extended gold span is acceptable, so we detect only single tokens for simplification. In order to simplify and restrict the number of tokens needed to classify, some heuristic restrictions are applied. We just consider those tokens having part-of-speech (POS) tags of noun, verb and adjective. Although triggers or entities have various POS tags, these three types take the largest proportion. Proteins in each sentence are replaced by a place holder "PROTEIN" instead of the original text. Those tokens related to protein (spans of a token and a protein are overlapped) are ignored. Instead we use a simple dictionary built from training data to check whether or not those tokens are triggers.

We classify tokens by their syntactic context and morphological contents. Features for detection include the candidate token; two immediate neighbors on both the left and right hand sides; POS tags of these tokens; and the nearest dependent and governor from the syntactic dependency path of the candidate token. All covered text used in classification is in lemmatized form.

2.3 Event detection

After trigger detection, we combined rule-based with feature-based classifiers for event detection. We first run the rule-base system and then continued to combine with SVM based using the output of the rule-based system in order to increase the performance of our system. At the SVM based phase, we generate features for all shortest dependency paths between predicted trigger and argument (protein or event). Each shortest path example is classified as positive and negative events. The overall best-performing system is the combination of all events of rule base and feature-based classifiers.

2.3.1 Rule-based approach

In this stage, rule-based approaches are applied. In order to add a supplement to our method, we attempt to combine two directions, bottom up and top down. Both of them use linguistic information, mostly syntactic and dependency graph. Two approaches are run separately; finally the two result sets are combined.

The first approach is based on patterns of syntactic graph. It follows the approach of (Björne et al., 2009), (Casillas et al., 2011). The original parse tree of each sentence containing at least one trigger is retrieved. Nodes with only one branch are pruned and the top node is kept to retain the most important parts. Concepts of candidate arguments (name role) and the trigger are assigned to appropriate tree-nodes according to their spans in the text. Next, we find the closest parent of all arguments. The patterns are the string form of the sub-tree of the modified parse tree. Then the patterns are compared with those extracted from training data.

The second approach considered a part of syntactic graph. Because of some similar properties between extracting events and protein-protein in-

[2] http://bllip.cs.brown.edu/resources.shtml

teractions (Bui et al., 2011), we construct some patterns connecting arguments and triggers. There are two kinds of patterns: noun phrases (NP) and verb phrases (VP). Each phrase has to have one trigger and at least one Protein. In the case of the NP, it contains two nouns without any other phrase or it includes a preposition phrase (PP) and the trigger has to be the head of this NP. In the second pattern, we find a VP which is a direct parent of the trigger. If there is a Protein in those phrases, we annotate an Event with the trigger and the Protein as core argument.

2.3.2 Feature-based classifier

For the featured-based classifier, we use a dictionary of pairs of trigger - trigger, pairs of trigger – protein and event triggers. These dictionaries are built from the training and development data. Additionally, we extract features for all shortest dependency paths between trigger and argument (protein or event) by using used in the work of (Björne et al., 2009) and (Maha Amami, 2012).

Element features: trigger/argument word, trigger/argument type and trigger/argument POS.

N-gram features: n-grams of dependencies, n-grams of words and n-gram of consecutive words representing governor-dependent relationship.

Frequency features: length of the shortest path between trigger and argument (protein or event), number of arguments and event triggers per type in the sentence.

Dependency features: Directions of dependency edges relative to the shortest path, types of dependency edges relative to the shortest path.

2.4 Post processing

In this section, we only scan all the annotated objects which are stored in the framework. Arguments of events are arranged and duplicated events are limited. Each valid detected Event Trigger/Entity and Event will be written into the result file according to the standard format of the shared task.

3 Experimental result

In order to perform evaluation, we implemented our event extraction system. Table 1 shows the latest results of our system as computed by the shared task organizers. We achieved an F-score of only 34.98%, ranked 10th among 10th participants and the result is far from satisfactory (the best result of the shared task 2013 is 50.97%). We need a better solution of post-processing step to improve performance and restrict unexpected results. Improving results of trigger detection also contributes to reduce false positive events. However, the gold data of the test set is not provided. It is therefore difficult to evaluate the effectiveness of the trigger annotation step and its impact on the event annotation step.

Event class	Recall	Precision	F-score
Gene_expression	78.84	61.77	69.27
Transcription	32.67	50.77	39.76
Protein_catabolism	64.29	52.94	58.06
Localization	32.32	52.46	40.00
Phosphorylation	77.50	57.67	66.13
Binding	38.74	26.99	31.81
Regulation	9.72	10.22	9.96
Positive_regulation	19.91	19.58	19.75
Negative_regulation	24.33	26.18	25.22
ALL-TOTAL	**36.23**	**33.80**	**34.98**

Table 1: Evaluation results on test set

4 Conclusion

In this paper we present an event extraction system based on combining rule-base with support vector machine modeling. Our system used the GENIA corpus as the input for the pre-processing phase such as Tokenization, Part-of-Speech, stop word removal and Stemming. In the trigger annotation, we extract the features for training and test data by using support vector machine classifier. In order to annotate events, firstly we use rule-based and then build the nested features using support vector machine classifier for event classification. The goal of this system is to increase the performance in F-score of the event extraction system.

In future work, we plan to try to add more features to improve our system both of trigger and event annotation and post-processing.

References

Casillas, A., Ilarraza, A.D., Gojenola, K., Oronoz, M., Rigau, G.: Using Kybots for Extracting Events in Biomedical Texts. In *Proceedings of BioNLP Shared Task 2011 Work-shop*, pp. 138-142. (2011).

Kilicoglu, H., Bergler, S.: Adapting a General Semantic Interpretation Approach to Bio-logical Event Extraction. In *Proceedings of BioNLP Shared Task 2009 Workshop*, pp. 173-182. (2011).

Bjorne, J., Heimonen, J., Ginter, F., Airola, A., Pahikkala, T., Salakoski, T.: Extracting Complex Biological Events with Rich Graph-Based Feature

Sets. *In Proceedings of BioNLP Shared Task 2009 Workshop*, pp. 10-18. (2009)

Kim, J.D., Ohta, T., Pyysalo, S., Kano, Y., Tsujii, J.: Overview of BioNLP'09 Shared Task on Event Extraction. In *Proceedings of BioNLP Shared Task 2011 Workshop*, pp. 1-9. (2011).

Kim, J.D., Wang, Y., Takagi, T., Yonezawa, A.: Overview of the Genia Event task in Bi-oNLP Shared Task 2011. In *Proceedings of BioNLP Shared Task 2011 Workshop*, pp. 7-15. (2011).

Kaljurand, K., Schneider, G., Rinaldi, F.: UZurich in the BioNLP 2009 Shared Task. In *Proceedings of BioNLP Shared Task 2009 Workshop*, pp. 28-36. (2009).

Miwa, M., Sætre, R., Kim, J.D., Tsujii, J.: Event Extraction with Complex Event Classification Using Rich Features. In *Journal of Bioinformatics and Computational Biology*, vol. 8, pp. 131-146. (2010).

Bui, Q.C., Sloot, P.M.A.: Extracting Biological Events from Text Using Simple Syntactic Patterns. In *Proceedings of BioNLP Shared Task 2011 Workshop*, pp. 143-146. (2011).

Le, M.Q., Nguyen, T.S., Ho, B.Q.: A Pattern Approach for Biomedical Event Annotation. In *Proceedings of BioNLP Shared Task 2011 Workshop*, pp. 149-150. (2011).

Riedel, S., McCallum, A.: Robust Biomedical Event Extraction with Dual Decomposition and Minimal Domain Adaptation. In *Proceedings of BioNLP Shared Task 2011 Work-shop*, pp. 46-50. (2011).

Riedel, S., McClosky, D., Surdeanu, M., McCallum, A., Manning, C.D.: Model Combination for Event Extraction in BioNLP 2011. In *Proceedings of BioNLP Shared Task 2011 Workshop*, pp. 51-55. (2011).

Maha Amami, Rim Faiz, Aymen Elkhlifi: A framework for biological event extraction from text. *Copyright 2012 ACM,* 978-1-4503-0915-8/12/06. WIMS' 12 June 13-15, 2012 Craiova, Romania