A fast rule-based approach for biomedical event extraction

Quoc-Chinh Bui
Department of Medical Informatics,
Erasmus Medical Centre
Rotterdam, Netherlands
q.bui@erasmusmc.nl

David Campos
IEETA/DETI, University of Aveiro
3810-193 Aveiro
Portugal
david.campos@ua.pt

Erik M. van Mulligen
Department of Medical Informatics,
Erasmus Medical Centre
Rotterdam, Netherlands
e.vanmulligen@erasmusmc.nl

Jan A. Kors
Department of Medical Informatics,
Erasmus Medical Centre
Rotterdam, Netherlands
j.kors@erasmusmc.nl

Abstract

In this paper we present a biomedical event extraction system for the BioNLP 2013 event extraction task. Our system consists of two phases. In the learning phase, a dictionary and patterns are generated automatically from annotated events. In the extraction phase, the dictionary and obtained patterns are applied to extract events from input text. When evaluated on the GENIA event extraction task of the BioNLP 2013 shared task, the system obtained the best results on strict matching and the third best on approximate span and recursive matching, with F-scores of 48.92 and 50.68, respectively. Moreover, it has excellent performance in terms of speed.

1 Introduction

A growing amount of biomedical data is continuously being produced, resulting largely from the widespread application of high-throughput techniques, such as gene and protein analysis. This growth is accompanied by a corresponding increase of textual information, in the form of articles, books and technical reports. In order to organize and manage these data, several manual curation efforts have been set up to identify entities (e.g., genes and proteins), their interactions (e.g., protein-protein) and events (e.g., transcription and gene regulation). The extracted information is then stored in structured knowledge resources, such as MEDLINE and Swiss-Prot. However, manual curation of large quantities of data is a very demanding and expensive task, and it is difficult to keep these databases up-to-date. These factors have naturally led to an increasing interest in the application of text mining (TM) systems to support those tasks.

Automatic recognition of biomedical events from scientific documents was highly promoted by the BioNLP challenges (Kim *et al.*, 2009; 2011), focusing on events that involve genes and proteins, such as gene expression, binding, and regulation. Such events are typically represented as the relation between a trigger and one or more arguments, which can be biomedical concepts or other events.

Several approaches have been proposed to extract biological events from text (Kim *et al.*, 2009; 2011). Based on their characteristics and applied natural language processing (NLP) tools, these approaches can be categorized into two main groups, namely rule- and machine learning (ML)-based approaches. Rule-based approaches consist of a set of rules that are manually defined or automatically learned from training data (Bui & Sloot, 2011; Cohen *et al*,. 2009; Kaljurand *et al*., 2009; Kilicoglu & Bergler, 2011). To extract events from text, first event triggers are detected using a dictionary, then the defined rules are applied to the output of the NLP tools e.g., dependency parse trees, to find their arguments. On the other hand, ML-based approaches exploit various feature sets and learning algorithms to extract events (Björne & Salakoski, 2011; Miwa *et al.*, 2010; 2012; Riedel & McCallum, 2011).

This article presents an enhanced version of our biomedical event extraction system (Bui & Sloot, 2012). Here we simplify the way patterns are generated from training data and improve the method to extract events from text based on the obtained patterns.

Proceedings of the BioNLP Shared Task 2013 Workshop, pages 104–108,
Sofia, Bulgaria, August 9 2013.

2 System and methods

The workflow of the system is illustrated in Figure 1. A text preprocessing step, which converts unstructured text into a structured representation, is applied for both learning and extraction phases. In the learning phase, a dictionary and patterns are generated automatically from annotated events. In the extraction phase, the dictionary and obtained patterns are applied to extract events from input text.

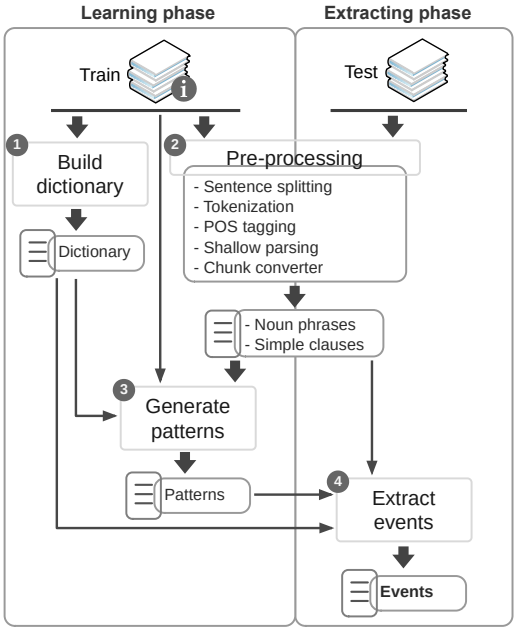

Figure 1: workflow of the system.

2.1 Text preprocessing

The text preprocessing step intends to break the input text into meaningful units, in order to reveal important linguistic features. This step consists of splitting input text into single sentences, tokenizing sentences, part-of-speech (POS) tagging, shallow parsing, and converting obtained chunks into simple clauses. An in-depth description of this step is provided in (Bui & Sloot, 2012). An example of a structured representation is illustrated in Figure 2.

2.2 Building a dictionary

The dictionary construction is carried out automatically using event triggers from training data. This process consists of four steps: grouping event triggers, calculating confidence scores, filtering out irrelevant triggers, and determining event types. First, we collect all event triggers annotated in the training dataset, convert them to lower-case and group them based on their text and event types. For each event trigger, we count the number of times it appears as an event trigger and the number of times it appears in the training dataset, in order to calculate its confidence score. Next, we filter out triggers that have POS tags not starting with "NN", "VB", or "JJ", as well as triggers that consist of more than two words, as suggested in a previous study (Kilicoglu & Bergler, 2011). We further filter out more triggers by setting a frequency threshold and confidence score for each event type. Finally, we assign an event type for each event trigger based on its type annotated in the training data. If an event trigger belongs to more than one event group, we determine its event type based on the event group where it appears with highest frequency. For instance, the "effect" trigger appears in both "Regulation" and "Positive_regulation" groups, but its frequency in the "Regulation" group is higher, therefore it is assumed to be a "Regulation" event trigger.

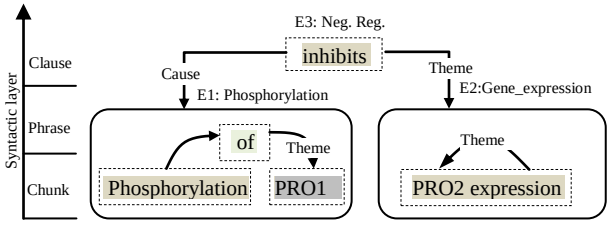

Figure 2: Structured representation of biomedical events.

2.3 Predefined patterns

When using a structured representation to express biomedical events, in most cases, an event can be mapped into a "container", i.e., a chunk, a phrase, or a clause as shown in Figure 2. Based on this representation, we define a list of the most common patterns that encode relations between an event trigger and its arguments. The predefined list of patterns is shown in Table 1. We skip all events that cannot be expressed within a simple clause.

Container	Pattern type
Chunk	Trg – Arg1
	Arg2-Trg-Arg1
	Arg1-Trg
Phrase	Trg-Prep1- Arg1
	Trg-Prep1-Arg1-Prep2 –Arg2
	Trg-Prep2-Arg2-Prep1 –Arg1
	Arg2-Trg-Prep1-Arg1
	Arg1-Arg2-Trg
Clause	Arg1 – Trg
	Trg – Arg1
	Arg2 – Trg – Arg1
	Arg1 – Trg – Arg2

Table 1: Common patterns for relations between an event trigger and its arguments. Trg denotes event trigger, prep: preposition, arg1: event theme, and arg2: theme2 or cause of an event.

2.4 Generating patterns

To generate a pattern for each event, first we find a suitable container (e.g., chunk, phrase, or clause) that contains the event trigger and its arguments. If such a container is found, a pattern is generated by extracting features from that container using a list of defined feature set as shown in Table 2. Each generated pattern is then assigned a key by combining its event trigger, POS tag, pattern type, and container type. This key is used to retrieve this pattern in the extraction step. During the learning process, if a key of a newly generated pattern already exists, the system increases the *frequency* attribute of the existing pattern and updates the other attributes accordingly.

Features	Description and examples
Trigger	Event trigger.
Prep1	Preposition between theme and trigger, e.g. *of, in.*
Pattern type	Defined in Table 1.
Prep2	Preposition between cause/theme2 and trigger.
Container	The container which contains this event.
Distance1	Distance (number of chunks) between theme and event trigger.
Distance2	Distance (number of chunks) between cause/theme2 and event trigger.
POS	POS tag of the trigger e.g. NN, ADJ, and VBZ.
Pro1 count	Count number of events with a protein as theme.
Even1 count	Count number of events with an event as theme.
Pro2 count	Count number of events with a protein as theme2/cause.
Even2 count	Count number of events with an event as theme2/cause.
Frequency	Number of events sharing the same pattern key. This value is used to rank the patterns in the extraction step.

Table 2: Feature set used to generate patterns.

2.5 Extracting events

In this step, we apply the obtained patterns to extract events from text. First, the input sentence is converted into a structured representation by applying the text preprocessing step. Next, tokens of each sentence are matched against the dictionary to detect candidate event triggers. For each candidate event trigger, a key is generated to retrieve its corresponding patterns. If patterns for the event trigger exist, we then apply the retrieved patterns using the order of the syntactic layers: chunk, phrase, and clause (see Figure 2). Furthermore, if there is more than one pattern available for a syntactic layer (e.g. chunk, phrase), the order to apply patterns is determined by the frequency of these patterns, which is calculated in the previous step. Patterns with higher frequency have higher priority.

3 Results

3.1 Datasets

We used the training and development datasets provided by the BioNLP'11 and BioNLP'13 shared tasks to train our system. The statistics of the datasets are presented in Table 3.

Items	Training	Test
Abstracts (+full papers)	950 (+20)	0 (+10)
Proteins	19089	4359
Events	16403	3301
Availability of events	Yes	Hidden

Table 3: Characteristics of the training and test datasets.

All training data were used to build the dictionary and generate patterns. In our experiment, we used the same dictionary for the learning and extraction phases. The confidence score of all entries in the dictionary was set to 0.1. In the extraction phase, the distance features ("Distance1" and "Distance2") were set to a maximum of 10 chunks, and patterns that have a frequency lower than 3 were not used in order to reduce false-positive events.

3.2 Event extraction

Table 4 presents the results achieved by our system on the BioNLP 2013 GENIA test dataset using both strict and approximate matching. Our system achieves an F-score of 48.92 with strict matching, and an F-score of 50.68 with approximate matching. For relaxed matching, the

data show that our system performs well on simple events ("simple all") with an average F-score of 76.11, followed by protein modification events ("prot-mod all") with an average F-score of 74.37. The performance declines on binding events with an F-score of 49.76 and regulatory events ("regulation all") with an average F-score of 35.80. When comparing the performance of our system between the two matching criteria, the data indicate that only *Transcription* events gain significant performance, with an F-score increase of 30 points.

Event type	Strict matching			Approximate span		
	R	P	F1	R	P	F1
Gene expression	72.86	85.74	78.78	73.83	86.88	79.83
Transcription	32.67	48.53	39.05	58.42	86.76	69.82
Protein catabolism	42.86	75.00	54.55	42.86	75.00	54.55
Localization	42.42	89.36	57.53	42.42	89.36	57.53
Simple all	**63.87**	**81.97**	**71.79**	**67.71**	**86.90**	**76.11**
Binding	47.45	52.32	49.76	47.45	52.32	49.76
Phosphorylation	82.50	80.49	81.48	82.50	80.49	81.48
Prot-mod all	**69.11**	**80.49**	**74.37**	**69.11**	**80.49**	**74.37**
Regulation	12.50	30.25	17.69	13.19	31.09	18.53
Positive regulation	30.62	49.93	37.96	31.68	51.66	39.28
Negative regulation	28.33	49.17	35.95	28.90	50.17	36.67
Regulation all	**27.31**	**47.62**	**34.72**	**28.19**	**49.06**	**35.80**
Event total	**40.99**	**60.67**	**48.92**	**42.47**	**62.83**	**50.68**

Table 4: Precision (P), recall (R) and F-score (F1) results achieved on the test set of BioNLP 2013, evaluated on strict matching and approximate span and recursive criteria.

Table 5 presents a comparison of the overall performance results with the top-five performing systems in the BioNLP 2013 GENIA task. The data show that our system (BioSem) achieves the best results on strict matching, and ranks third on approximate matching, with a slight difference in F-score of 0.29 point compared to the best system. Furthermore, our system yields the best precision on both matching criteria, with a considerable difference on strict matching.

Team	Strict matching			Approximate span		
	R	P	F1	R	P	F1
EVEX	42.99	54.89	48.22	45.44	58.03	**50.97**
TEES-2.1	**43.71**	53.33	48.04	**46.17**	56.32	50.74
NCBI	37.35	56.72	45.04	40.53	61.72	48.93
DlutNLP	37.75	52.73	44.00	40.81	57.00	47.56
BioSem	40.99	**60.67**	**48.92**	42.47	**62.83**	50.68

Table 5: Performance comparison of overall Precision (P), recall (R) and F-score (F1) with the five best systems.

A closer look at the official results (data not shown) reveals that our system obtains the best performance on Binding event with an F-score of 49.76, which is significantly higher than the second-best system (F-score 43.32). Interestingly, our system also yields the highest F-score (58.77) when evaluated on themes only.

When aiming for a large-scale relation extraction, system performance in terms of speed has to be taken into account. By employing a simple text processing and an effective event extraction algorithm, our system is very fast. On a standard PC with 4GB of RAM, it takes 49s to process the training dataset and 11s to process the test dataset.

4 Conclusion and future work

This article presents a system for biomedical event extraction that generates patterns automatically from training data. When evaluated on the test set, it presented the best results with strict matching and the third best with approximate span and recursive matching. Moreover, it obtains high precision on both evaluation criteria, and has an excellent performance in terms of speed.

There are various ways to further improve the performance of the system. First, we believe that an ML-based approach for trigger recognition will improve its results, by minimizing ambiguity problems and improving recall, especially on regulatory events. Second, the final performance depends on the output of the text-preprocessing step, especially the conversion of chunks into structured representations. If the performance of this step is improved, for example by using predicate argument structures as proposed by (Miwa *et al.*, 2010) to obtain relations between subject-verb-object, then more precise patterns could be obtained in the learning phase. Consequently, the extraction phase would have a cleaner input (with less false positives and false negatives), which will eventually enhance the performance. Furthermore, as proposed in our previous study (Bui *et al.*, 2011), the output of the current system can be used as the input for an ML classifier to further reduce false-positive events. The feature set used in the predefined patterns can also be used directly as feature set for the ML classifier.

Acknowledgments

D. Campos was funded by FEDER through the COMPETE programme and by national funds through FCT - "Fundação Para a Ciência e a Tecnologia" under the project number PTDC/EIA-CCO/100541/2008.

References

Björne, J., & Salakoski, T. (2011). Generalizing biomedical event extraction (pp. 183–191). Presented at the BioNLP Shared Task 2011 Workshop, Portland, Oregon, USA: Association for Computational Linguistics.

Bui, Q. C., & Sloot, P. (2011). Extracting biological events from text using simple syntactic patterns (pp. 143–146). Presented at the BioNLP Shared Task 2011 Workshop, Portland, Oregon, USA.

Bui, Q.-C., & Sloot, P. M. A. (2012). A robust approach to extract biomedical events from literature. *Bioinformatics (Oxford, England)*, *28*(20), 2654–2661. doi:10.1093/bioinformatics/bts487

Bui, Q.-C., Katrenko, S., & Sloot, P. M. A. (2011). A hybrid approach to extract protein-protein interactions. *Bioinformatics (Oxford, England)*, *27*(2), 259–265.

Cohen, K. B., Verspoor, K., Johnson, H. L., Roeder, C., Ogren, P. V, Jr, W. A. B., White, E., et al. (2009). High-precision biological event extraction with a concept recognizer. Proceedings of BioNLP'09 Shared Task Workshop (pp. 50–58).

Kaljurand, K., Schneider, G., & Rinaldi, F. (2009). UZurich in the BioNLP 2009 shared task. Proceedings of BioNLP'09 Shared Task Workshop (pp. 28–36).

Kilicoglu, H., & Bergler, S. (2011). Adapting a general semantic interpretation approach to biological event extraction (pp. 173–182). Presented at the BioNLP Shared Task 2011 Workshop, Portland, Oregon, USA: BioNLP Shared Task 2011 Workshop.

Kim, J.-D., Ohta, T., Pyysalo, S., Kano, Y., & Tsujii, J. (2009). Overview of BioNLP'09 shared task on event extraction (pp. 1–9). Presented at the BioNLP Shared Task 2009 Workshop, Boulder, Colorado, USA: Association for Computational Linguistics.

Kim, J.-D., Wang, Y., Takagi, T., & Yonezawa, A. (2011). Overview of genia event task in bionlp shared task 2011 (pp. 7–15). Presented at the BioNLP Shared Task 2011 Workshop, Portland, Oregon, USA: Association for Computational Linguistics.

Miwa, M., Sætre, R., Kim, J.-D., & Tsujii, J. (2010). Event extraction with complex event classification using rich features. *Journal of bioinformatics and computational biology*, *8*(1), 131–146.

Miwa, M., Thompson, P., & Ananiadou, S. (2012). Boosting automatic event extraction from the literature using domain adaptation and coreference resolution. Bioinformatics (Oxford, England), 28(13), 1759–65.

Riedel, S., & McCallum, A. (2011). Robust biomedical event extraction with dual decomposition and minimal domain adaptation. Presented at the BioNLP Shared Task 2011 Workshop, Portland, Oregon, USA.