LitWay, Discriminative Extraction for Different Bio-Events

Chen Li
Xi'an Jiaotong University, China
Massachusetts Institute of
Technology, United States
`cli@xjtu.edu.cn`

Zhiqiang Rao
Xidian University, China
`zhiqiangrao`
`@foxmail.com`

Xiangrong Zhang
Xidian University, China
Massachusetts Institute of
Technology, United States
`xrzhang`
`@mail.xidian.edu.cn`

Abstract

Even a simple biological phenomenon may introduce a complex network of molecular interactions. Scientific literature is one of the trustful resources delivering knowledge of these networks. We propose LitWay, a system for extracting semantic relations from texts. LitWay utilizes a hybrid method that combines both a rule-based method and a machine learning-based method. It is tested on the SeeDev task of BioNLP-ST 2016, achieves the state-of-the-art performance with the F-score of 43.2%, ranking first of all participating teams. To further reveal the linguistic characteristics of each event, we test the system solely with syntactic rules or machine learning, and different combinations of two methods. We find that it is difficult for one method to achieve good performance for all semantic relation types due to the complication of bio-events in the literatures.

1 Introduction

Bio-events are founding blocks of bio-networks depicting profound biological phenomena. Automatically extracting bio-events may assist researchers while facing the challenge of growing amount of biomedical information in textual form. A bio-event carries more semantic information biochemical reactions between entities, therefore, is more informative for studying associations between bio-concepts, e.g. gene and phenotype (Li et al., 2013).

A number of methods have been proposed to process the automated extraction of biomedical events including rule-based (Cohen et al., 2009; Kilicoglu and Bergler, 2011; Bui and Sloot, 2011)

and machine learning-based (Miwa et al., 2012; Hakala et al., 2013; Munkhdalai et al., 2015) methods. Bui et al. (2013) presented a rule-based method for bio-event extraction by using a dictionary and patterns generated automatically from annotated events. TEES (Björne and Salakoski, 2013) is a SVM based text mining system for the extraction of events and relations from natural language texts, it obtains good performance on a few tasks in BioNLP-ST 2013 (Nédellec et al., 2013). As a major type of biomedical events, a series of methods concentrate on protein-protein interactions (PPI) (Miyao et al., 2009; Papanikolaou et al., 2015). Kernel-based methods are widely used for relation extraction task and obtain good results by leveraging lexical and syntactic information (Airola et al., 2008; Miwa et al., 2009; Li et al., 2015b). Peng et al. (2015) proposed Extended Dependency Graph (EDG) and evaluated it with two kernels on some PPI datasets, obtained good improvements on F-value.

We previously use a set of basic features including word embedding on a classifier for the BioNLP 2013 Genia (Kim et al., 2013) dataset, the result is comparable to the state-of-the-art solution (Li et al., 2015a). The system is built with flexibility in mind. It is designed to tackle more types of bio-events. In this paper, we introduce LitWay, which is based on the previous infrastructure and uses a machine learning based method in combination with syntactic rules. The system is tested on a completely different task, the SeeDev of BioNLP-ST 2016. It achieves the best result among all participants with an F-score of 43.2% (recall and precision are 44.8% and 41.7% respectively).

2 SeeDev Task

As a popular task in unstructured data mining of biomedical interests, BioNLP has successfully
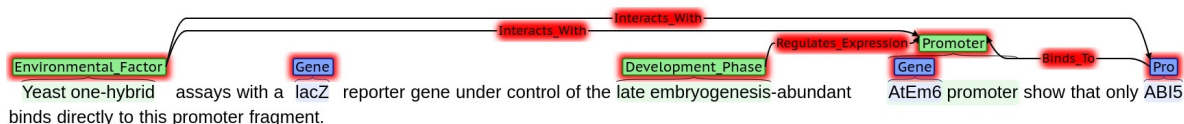
Figure 1: Event relation examples. This sentence includes 4 events and 6 entities. For example, a Environmental_Factor *Yeast one-hybrid* and a Protein *ABI5* form a event Interacts_With. An entity could participate in several events at the same time or none, such as *AtEm6 promoter* and *lacZ*. Noticeably entity span overlap, like Gene *AtEm6* and Promoter *AtEm6 promoter*.

held a series of biomedical event extraction tasks. GE (Genia Event Extraction) is a classic task initiated since the beginning of BioNLP (Kim et al., 2009), it attracts attention and leads to abundant works (Kim et al., 2011; Kim et al., 2013). Be similar to GE and others of BioNLP, SeeDev (Chaix et al., 2016) is a new task proposed in BioNLP-ST 2016, it dedicates to event extraction of genetic and molecular mechanisms involved in plant seed development. It is based on the knowledge model Gene Regulation Network for Arabidopsis (GR-NA)[1]. GRNA model defines 16 different types of entities, and 22 event types that may be combined in complex events. Table 1 shows these entities. Event types are presented in following. Figure 1 gives some examples of event relations[2].

Entity type	
Gene	Protein_Domain
Gene_Family	Hormone
Box	Regulatory_Network
Promoter	Pathway
RNA	Genotype
Protein	Tissue
Protein_Family	Development_Phase
Protein_Complex	Environmental_Factor

Table 1: 16 different entity types.

3 Proposed Method

LitWays pipeline adopts a hybrid method that uses a classifier or rule-based method for different event types. Figure 2 shows the infrastructure of it. The pipeline consists of 5 steps: pre-processing, entity pair selection, feature extraction, classifier prediction and rule-based filters.

In BioNLP-ST 2013, the top three event extraction systems F-scores differ less than 0.3%
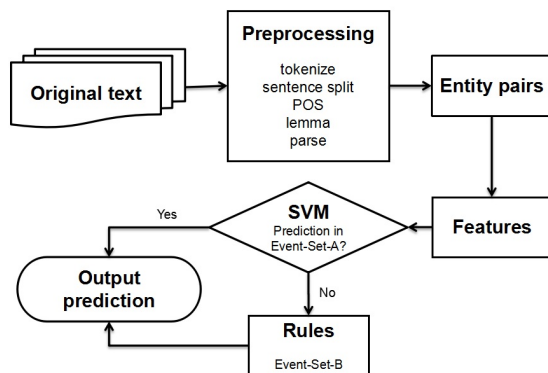

Figure 2: Infrastructure of LitWay.

in number (Nédellec et al., 2013; Björne and Salakoski, 2013). Differences of quantitative and syntactic morphology of proteins and chemical entities in the scientific literature might demand different strategies of network extraction to achieve a better performance. In this paper, we utilize a flexible hybrid system to investigate a way to discriminatively treat event types. We first pre-experiment on the development data and divide all event types into two sets: Event-Set-A and Event-Set-B. The events showing better performance on SVM are classified into Event-Set-A, the others showing better results on a rule-based method are classified into Event-Set-B. Event-Set-A composes of minority events, Event-Set-B composes of majority events except two types: Composes_Primary_Structure and Composes_Protein_Complex[3]. Two sets are showed in Table 2. It is easier to create precise and useful rules for majority events since there are enough instances for analyzing. Compared with using SVM for all events, better results are obtained from the experiment by using the rules.

---

[1]See details at http://2016.bionlp-st.org/tasks/seedev.

[2]From training dataset: SeeDev-binary-11489176-1.

[3]After the analysis with experiment results, while moving Composes_Primary_Structure and Composes_Protein_Complex into Event-Set-A, a slightly better F-score on all events could be obtained.

Event-Set-A	Number
Is_Linked_To	44
Regulates_Accumulation	36
Transcribes_Or_Translates_To	25
Is_Involved_In_Process	23
Occurs_In_Genotype	18
Regulates_Molecule_Activity	16
Exists_At_Stage	15
Regulates_Tissue_Development	9
Occurs_During	8
Event-Set-B	**Number**
Regulates_Process	436
Regulates_Expression	201
Exists_In_Genotype	169
Is_Localized_In	107
Regulates_Development_Phase	106
Is_Member_Of_Family	89
Has_Sequence_Identical_To	62
Interacts_With	62
Is_Functionally_Equivalent_To	60
Binds_To	60
Is_Protein_Domain_Of	46
Composes_Primary_Structure	20
Composes_Protein_Complex	16

Table 2: Event-Set-A and Event-Set-B. This partition was used during competition.

After pre-processing the raw text data, candidate entity pairs are constructed within each sentence, and tested by a multiclass classifier. If the classifier predicts that a candidate pair is a event belonging to Event-Set-A, the predication stays. Otherwise, a series of rules are used for deciding a type in Event-Set-B.

3.1 Pre-processing

The pre-processing include tokenization, sentence splitter, part-of-speech (POS) tagging, lemmatization and syntactic parsing. Stanford CoreNLP tool (Manning et al., 2014) is adopted for the operations.

3.2 Entity Pair Selection

The system aims to resolve semantic relation extraction as expected by the SeeDev task. In the task, each event has two arguments. We construct two entities as a candidate pair each time and predict their relation type. Table 3 presents sentence distance statistics of events on the training set, nearly 96.5% of events span within one sentence.

Since most events occur within a sentence, we only choose entity pairs in the same sentence.

Except three event types Is_Linked_To, Has_Sequence_Identical_To, and Is_Functionally_Equivalent_To, in which two arguments could be reversed, for the others they are ordered. Therefore an entity pair (Entity1, Entity2) is different from the reversed pair (Entity2, Entity1). They should be treated as two instances.

Sentence distance	Number
0	1571
1	52
2	5

Table 3: Sentence distance statistics of events on the training set.

3.3 Feature Extraction

The features are extracted and summarized in Table 4, which shows two types of features, entity features and entity pair features.

Entity feature	Entity pair feature
Entity type	Tree path
Words	Tree path length
Lemmas	Token distance
POSs	Entity distance
Unigram word	Middle lemmas
Unigram lemma	
Unigram POS	
Tree node depth	
Average word embedding	

Table 4: Features used in classifier. Entity features are extracted from two entities, separately. Entity pair features are extracted from a pair of entities.

Word, lemma, Part-Of-Speech (POS) are features directly represent an entity's lexical and grammatical characteristics. Adjacent words' features are used to represent the entity's contextual characteristics. Therefore, basic features include word, lemma, POS of entities, as well as the same information of the unigram words.

Generally speaking, if two entities are closer, they are more likely to be relative (Tikk et al., 2013). Token distance and entity distance are used here. Token distance is the number of tokens between two entities. Entity distance is the number of entities in the middle of two entities.

Syntactic parsing tree features are important for semantic relation (Punyakanok et al., 2008; D'Souza and Ng, 2012). Tree node depth, tree path, tree path length are used in our experiment. They are obtained from the syntactic parsing tree, generated during the pre-processing. Tree node depth is the distance between the corresponding tree node of an entity and the root node of the sentence. Tree path is the path between two entities. Tree path length is the number of middle nodes between two entities in their tree path.

Word embedding has demonstrated the ability of well representing linguistic and semantic information of a text unit (Mikolov et al., 2013; Tang et al., 2014), e.g. POS and N-gram. We continue using it as a feature in our system. Specifically, training, development and test datasets of SeeDev are used to obtain word embedding by using word2vec tool (Mikolov et al., 2013) after sentencization, tokenization and lemmatization on the original text. Since the word number of an entity is uncertain, we use the average value of all the word embeddings of an entity (Chen et al., 2015; Wang et al., 2015), i.e. average word embedding. Middle lemmas include all of the lemmas between two entities, they are treated as a bag-of-word (BOW) feature, some keyword information may be obtained from it.

3.4 SVM Classifier Prediction

Support Vector Machine (SVM) (Cortes and Vapnik, 1995) and the C++ embodiment, LibSVM (Chang and Lin, 2011), is employed for the classification in LitWay. Positive event instances are retrieved from gold annotations. Negative instances are created by all of no-relation entity pairs within each sentence.

Among predication, if the predicted result of an entity pair belongs to Event-Set-A, it is taken as the label. Otherwise, rule-based filters are applied.

3.5 Rule-based Filters

In Event-Set-B, different event types have different rules. We summarize all rules in Table 5. We consider the event types of Event-Set-B one by one, according to their quantities on the training set, as showed in Table 2. Once all rules of an event type are satisfied, the entity pair label could be determined, and the matching of the rest event types could be stopped.

There are 6 types of rules: Event arguments match, Entity structure rules, Sentence structure rules, Token distance restriction, Keywords match and Training set match. The details about these rules are shown as following:

(1) **Event arguments match**: According to the task description, the arguments of the event are strongly typed, which means that all types of entities are not possible as event arguments. What is more, according to the statistics of arguments of different events on the training set, we only retain those arguments that occur most times for each special event type. This could efficiently reduce false instances.

(2) **Entity structure rules**: Many entities have complicated structure, an entity could span over another entity. This results in that some entity structures are less likely to be event arguments. Such as, an entity with smaller span is not an argument, as it is often the modifier of the larger one. Meanwhile, some event types have several fixed special entity argument structures. We summarize 3 particular rules from the training set:

- (2a) Entity is not covered: An entity is not covered by a larger one.

- (2b) Entity does not cover: An entity does not contain smaller entities or overlap with others.

- (2c) Special entity structure: Some special entity structure rules are summarized from the dataset. Presumably an entity pair (Entity1, Entity2) is within a sentence, Entity is another entity in the same sentence, the special entity structures could be:

  - (2c1) *Entity1 (Entity2)*: Entity pair should have such fixed special structure, Entity2 follows Entity1 and is in brackets.
  - (2c2) *Entity1 (Entity)*: If Entity1 is followed by Entity and Entity is in brackets, Entity1 is ignored.
  - (2c3) *Entity (Entity1)*: If Entity1 follows Entity and Entity1 is in brackets, Entity1 is ignored.
  - (2c4) *Entity2 Entity1 (Entity)*: If Entity follows Entity1 and Entity is in brackets, while Entity1 also follows Entity2, then Entity1 is kept.
  - (2c5) *Entity (Entity2)*: If Entity2 follows Entity and Entity2 is in brackets, Entity2 is ignored.

(3) **Sentence structure rules**: If two entities form an event relation, the sentence structure presents some syntactic characteristics. We summarize 3 sentence structure rules:

- (3a) No subordinate clause: Subordinate clause is a complex sentence structure. If there is event relation between a pair of entities, the syntactic tree path structure between them is often simple and direct.

- (3b) Active or passive structure match: For an event argument pair (Entity1, Entity2), it should have such relation structure: *Entity1-influences-Entity2*. While an entity pair has two orders in a sentence: Entity1 is on the left of Entity2, or right of Entity2. Different orders should match different sentence structure rules. If Entity1 is on the left of Entity2, their tree path is an active structure. Otherwise it is a passive structure.

- (3c) Special entity pair order: Some events usually have fixed order between their two arguments, Entity1 is always on the left (or right) of Entity2.

(4) **Token distance restriction**: Closer entities are more likely to be relative. The rule restricts the number of middle tokens between entities. It ignores distant entity pairs.

(5) **Keywords match**: Some events are accompanied by keywords, we record these keywords of several different events, showed in following detailed rules. They are useful for event identification.

(6) **Training set match**: For some event types, we compile the entity pairs from the training set into a dictionary, since they are biologically more likely to interact.

For an entity pair (Entity1, Entity2), we apply the rules on Event-Set-B. Following labels (1) to (6) correspond to 6 type rules introduced above, *None* means nonuse of this rule:

Regulates_Process
(1) Entity1∈{Genotype, Tissue, Gene, Protein, Development_Phase},
Entity2∈{Regulatory_Network, Pathway}
(2) Entity1 is not covered
(3) No subordinate clause, Active or passive structure match
(4) None
(5) None

(1) Event arguments match
(2) Entity structure rules
• (2a) Entity is uncovered
• (2b) Entity does not cover
• (2c) Special entity structure
(3) Sentence structure rules
• (3a) No subordinate clause
• (3b) Active or passive structure match
• (3c) Special entity pair order
(4) Token distance restriction
(5) Keywords match
(6) Training set match

Table 5: Summary of all rules.

(6) None
Regulates_Expression
(1) Entity1∈{Tissue, Genotype, Protein, Development_Phase}, Entity2∈{Gene}
(2) Entity1 is not covered, Entity2 is not covered
(3) No subordinate clause
(4) None
(5) Keywords∈{function, target, repress, bind, regulat-, exclude, activate, require, expression, induce, detect, express, define, act, during, plicate, observe, affect, defect, transcription, cease, associate, restrict, modulate}
(6) None
Exists_In_Genotype
(1) Entity1∈{Gene, Gene_Family, RNA, Protein, Protein_Family, Protein_Domain}
Entity2∈{Genotype}
(2) Entity1 is not covered, Entity2 does not cover
(2c2) *Entity1 (Entity)*: ignore Entity1
(2c4) *Entity2 Entity1 (Entity)*: keep Entity1
(2c5) *Entity (Entity2)*: ignore Entity2
(3) No subordinate clause
(4) None
(5) None
(6) None
Is_Localized_In
(1) Entity1∈{RNA, Protein, Protein_Family, Protein_Complex, Protein_Domain, Hormone}
Entity2∈{Tissue}
(2) Entity1 is not covered, Entity2 is not covered
(3) No subordinate clause
(4) None
(5) None
(6) None
Regulates_Development_Phase
(1) Entity1∈{Gene, Protein, Genotype,

Gene_Family}, Entity2∈{Development_Phase}
(2) Entity1 is not covered, Entity2 is not covered
(2c2) *Entity1 (Entity)*: ignore Entity1
(3) Entity1 is on the left of Entity2
(4) None
(5) None
(6) None

Is_Member_Of_Family
(1) (Entity1, Entity2)∈{(Protein, Protein_Family), (Gene, Gene_Family)}
(2) Entity1 is not covered, Entity2 is not covered
(2c2) *Entity1 (Entity)*: ignore Entity1
(3) No subordinate clause
(4) Token distance ≤ 10
(5) None
(6) None

Has_Sequence_Identical_To
(1) Entity1 and Entity2 have same entity type
(2) (2c1) *Entity1 (Entity2)*: Entity pair has this structure
(3) None
(4) Token distance = 2
(5) None
(6) Training set match

Interacts_With
(1) Entity1∈{Protein, Environmental_Factor}, Entity2∈{Box, Promoter, Protein, Protein_Family, Protein_Complex, Protein_Domain}
(2) Entity1 is not covered, Entity2 is not covered
(3) No subordinate clause
(4) None
(5) Keywords∈{interacts, interacted, interacting, associate, associated, associates, associating}
(6) None

Is_Functionally_Equivalent_To
(1) Entity1 and Entity2 have same entity type
(2) (2c1) *Entity1 (Entity2)*: Entity pair has this structure
(3) None
(4) Token distance = 2
(5) None
(6) Training set match

Binds_To
(1) Entity1∈{Protein, Protein_Family, Protein_Domain}, Entity2∈{Box, Promoter, Protein, Protein_Family, Protein_Complex}
(2) Entity1 is not covered, Entity2 is not covered
(3) No subordinate clause
(4) None
(5) Keywords∈{bind, binds, interact, physical, direct}

(6) None
Is_Protein_Domain_Of
(1) Entity1∈{Protein_Domain},
Entity2∈{Protein, Protein_Family}
(2) Entity1 is not covered, Entity2 is not covered
(2c2) *Entity1 (Entity)*: ignore Entity1
(3) None
(4) None
(5) None
(6) None

Composes_Primary_Structure
(1) Entity1∈{Box}, Entity2∈{Gene, Box, Promoter}
(2) None
(3) No subordinate clause
(4) None
(5) None
(6) None

Composes_Protein_Complex
(1) Entity1∈{Protein},
Entity2∈{Protein_Complex}
(2) Entity1 is not covered, Entity2 is not covered
(3) No subordinate clause, Entity1 is on the left of Entity2
(4) None
(5) None
(6) None

4 Results

To investigate the impact of different strategies and their comparison with the hybrid method, we test the system solely with machine learning, syntactic rules, or different combinations of them.

We compared the proposed hybrid method with the classifier-only based method on the development dataset. Table 6 shows the experiment results. All of the features are beneficial for the classifier, by using all of them we get the best SVM based result with 31.5% F1. Tree features make most improvement with 5.7% increase on F1, both recall and precision are increased. Dist features make only 0.2% F1 improvement and WM features make 1.2% F1 improvement. They increase precision with the loss of recall, while Tree features mainly contribute to recall.

Comparing hybrid method with the best SVM result in Table 6, we could see an obvious advantage. The F1 of the hybrid method is over 10% higher than the best SVM result, it greatly improves recall with around 16%, and has 3.4% precision increase. It's interesting because adding

Method	F1	R	P
(1) Word+POS+Lemma	0.244	0.206	0.300
(2) WPL+Dist	0.246	0.192	0.344
(3) WPL+Dist+Tree	0.303	0.267	0.348
(4) WPL+Dist+Tree+WM	0.315	0.264	0.390
(5) Hybrid	0.423	0.423	0.424

Table 6: Comparison between different features for SVM on development dataset. Methods (1) to (4) only use multiclass SVM with different feature selections, (5) is the hybrid method. WPL are word, POS and lemma features. Dist are token distance and entity distance features. Tree are tree node depth, tree path and tree path length features. WM are average word embedding and middle lemmas features.

rules usually increase precision instead of recall.

To verify the effect of rule-based method for different event types, we take the best SVM result as a basis, and then replace each event type with rule-based method in turns. Event-Set-B uses specific rules introduced before. Event-Set-A uses some frequent rules from Event-Set-B since it is difficult to create precise rules for minority class, they include:
(1) Event arguments match;
(2) Entity1 is not covered, Entity2 is not covered;
(3) No subordinate clause, active or passive structure match.

Table 7 presents the results. Except for Composes_Primary_Structure and Composes_Protein_Complex, F1 of Event-Set-B events are increased by using rules instead of SVM. While rules are not helpful for Event-Set-A, it verifies the partition of two sets.

Since Composes_Primary_Structure and Composes_Protein_Complex have better results in SVM, we move them into Event-Set-A and indeed get a little better result in overall events after the competition, it is showed in following.

Table 8 presents the details of SVM method and hybrid method. Almost all the events of Event-Set-B have better results in the hybrid method. This demonstrates the effectiveness of it.

To investigate the rules used in the proposed method, we take several experiments on the development data by different rule combinations. Table 9 presents their results. All of these rules are beneficial to the system more or less. Event arguments match and entity structure rules have important in-

Method	F1
WPL+Dist+Tree+WM	0.315
Regulates_Process	0.334
Regulates_Expression	0.315
Exists_In_Genotype	0.355
Is_Localized_In	0.323
Regulates_Development_Phase	0.325
Is_Member_Of_Family	0.320
Has_Sequence_Identical_To	0.328
Interacts_With	0.323
Is_Functionally_Equivalent_To	0.330
Binds_To	0.316
Is_Protein_Domain_Of	0.327
Composes_Primary_Structure	0.313
Composes_Protein_Complex	0.313
Is_Linked_To	0.248
Regulates_Accumulation	0.289
Transcribes_Or_Translates_To	0.315
Is_Involved_In_Process	0.306
Occurs_In_Genotype	0.314
Regulates_Molecule_Activity	0.277
Exists_At_Stage	0.313
Regulates_Tissue_Development	0.302
Occurs_During	0.310

Table 7: Replace each event type with rule-based method in turns on the basis of SVM. Event-Set-B is in italic.

fluences to the performance, result in around 10% and 8% F1 decrease respectively. It is understandable because almost all kinds of event types in Event-Set-B use these two rules, which makes them important to the system, especially on the precision. Sentence structure rules and keywords match are also useful, around 3% to 3.5% F1 improvement could be obtained by using them. They improve the performance by increasing the precision of the system with the loss of recall. Token distance restriction and training set match have only 0.1% to 0.3% influences on F1 as they are merely used in one or two event types. Token distance restriction could improve the precision while training set match improves the recall.

Table 10 is the official result of the SeeDev task (Chaix et al., 2016). LitWay achieves the best result among all participating teams with 43.2% F1 showing significant advantage. The recall of LitWay is 44.8%, which is comparable to the highest recall 45.8%. Its precision 41.7% is the second highest value, only lower than 53.3%.

Event type	WPL+Dist+Tree+WM			Hybrid		
	F1	Recall	Precision	F1	Recall	Precision
All	0.315	0.264	0.390	0.423	0.423	0.424
Regulates_Process	0.437	0.447	0.428	0.511	0.520	0.503
Regulates_Expression	0.416	0.387	0.448	0.432	0.360	0.541
Exists_In_Genotype	0.224	0.210	0.239	0.540	0.630	0.472
Is_Localized_In	0.433	0.447	0.420	0.518	0.617	0.446
Regulates_Development_Phase	0.182	0.136	0.276	0.333	0.424	0.275
Is_Member_Of_Family	0.342	0.255	0.519	0.479	0.418	0.561
Has_Sequence_Identical_To	0.514	0.429	0.643	0.807	0.735	0.893
Interacts_With	0.080	0.063	0.111	0.301	0.344	0.268
Is_Functionally_Equivalent_To	0.467	0.318	0.875	0.667	0.590	0.767
Binds_To	0.177	0.125	0.300	0.256	0.208	0.333
Is_Protein_Domain_Of	0.061	0.035	0.250	0.455	0.517	0.405
Composes_Primary_Structure	NA	0	NA	0.238	0.667	0.145
Composes_Protein_Complex	NA	NA	NA	NA	NA	0
Is_Linked_To	0.118	0.087	0.182	0.118	0.087	0.182
Regulates_Accumulation	0.271	0.207	0.429	0.271	0.207	0.429
Transcribes_Or_Translates_To	0.174	0.154	0.200	0.174	0.154	0.200
Is_Involved_In_Process	NA	0	0	NA	0	0
Occurs_In_Genotype	NA	0	NA	NA	0	NA
Regulates_Molecule_Activity	NA	NA	NA	NA	NA	NA
Exists_At_Stage	NA	0	0	NA	0	0
Regulates_Tissue_Development	NA	0	NA	NA	0	NA
Occurs_During	NA	0	NA	NA	0	NA

Table 8: Detailed results of SVM classifier and hybrid method on development dataset. *NA* in Recall represents none of this class instance on the development data. *NA* in Precision represents none of this class instance in the predicted results. *0* in Recall or Precision means none of True Positive (TP) instance of this type is obtained in the predicted results. Event-Set-B is in italic.

We present two more additional experiments after the competition by moving Composes_Primary_Structure and Composes_Protein_Complex into Event-Set-A. Table 11 shows the results. The result on development data has 0.8% improvement on F1, while does not show benefit on test data.

We analyse the results on development dataset before and after the movement operation. Before the movement, for Composes_Primary_Structure there are 10 True Positive (TP) instances among 69 predicted instances (gold number is 15), for composes_Protein_Complex there are 0 TP instance among 8 predicted result (gold number is 0). After the operation both of the two predicted numbers are 0, i.e. we do not make any predictions of the two event types. In this case, 10 right events are lost, on the other hand 67 false events are successfully deleted. It brings more benefits than harm.

5 Conclusion

The paper presents a hybrid method system Lit-Way, to resolve the biomedical semantic relations. It achieves the best result in BioNLP-ST 2016 SeeDev task. It is built as a flexible way with the awareness of that different bio-events have different linguistic characteristics and are difficult to be tackled by a single method.

Without much feature engineering nor complex algorithm, LitWay obtains the state-of-the-art performance on the official test data, with the highest F-score 43.2%. A series of experiments of using the methods and their combinations are carried out to investigate the different linguistic characteristics of different event types.

Here we extract relations within one sentence. While a number of events still span across sentences. By incorporating coreference technics in the future, we expect to be able to interconnect

Method	F1	R	P
(1) Hybrid	0.423	0.423	0.424
(2) No arguments match	0.325	0.484	0.244
(3) No entity rules	0.342	0.447	0.277
(4) No sentence rules	0.394	0.501	0.325
(5) No token distance	0.420	0.423	0.417
(6) No keywords	0.388	0.441	0.347
(7) No training match	0.422	0.420	0.424

Table 9: Hybrid experiment results with different rules on development dataset. Methods (2) to (7) have been removed one type rule separately on the basis of (1). Method (2) only follows the event argument match rules given by the SeeDev task (`http://2016.bionlp-st.org/tasks/seedev/seedev-data-representation.`), does not filter event arguments that never or rarely occur.

Method	F1	Recall	Precision
LitWay	0.432	0.448	0.417
UniMelb	0.364	0.386	0.345
VERSE	0.342	0.458	0.273
–	0.335	0.245	0.533
ULISBOA	0.306	0.256	0.379
LIMSI	0.255	0.318	0.212
DUTIR*	–	–	–

Table 10: Official evaluation results on test data.

events at the same time improve the event extraction performance.

References

Antti Airola, Sampo Pyysalo, Jari Björne, Tapio Pahikkala, Filip Ginter, and Tapio Salakoski. 2008. All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning. *BMC bioinformatics*, 9(11):1.

Jari Björne and Tapio Salakoski. 2013. Tees 2.1: Automated annotation scheme learning in the bionlp 2013 shared task. In *Bionlp Shared Task 2013 Workshop*, pages 16–25.

Quoc-Chinh Bui and Peter Sloot. 2011. Extracting biological events from text using simple syntactic patterns. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 143–146. Association for Computational Linguistics.

Quoc-Chinh Bui, David Campos, Erik van Mulligen, and Jan Kors. 2013. A fast rule-based approach for biomedical event extraction. In *Proceedings of the*

Dataset	F1	Recall	Precision
development	0.431	0.410	0.453
test	0.432	0.439	0.426

Table 11: Additional experiment results.

BioNLP Shared Task 2013 Workshop, pages 104–108. Association for Computational Linguistics.

Estelle Chaix, Bertrand Dubreucq, Abdelhak Fatihi, Dialekti Valsamou, Robert Bossy, Mouhamadou Ba, Louise Deléger, Pierre Zweigenbaum, Philippe Bessières, Loïc Lepiniec, and Claire Nédellec. 2016. Overview of the regulatory network of plant seed development (seedev) task at the bionlp shared task 2016. In *Proceedings of the 4th BioNLP Shared Task workshop*, Berlin, Germany, August. Association for Computational Linguistics.

Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Xinxiong Chen, Lei Xu, Zhiyuan Liu, Maosong Sun, and Huanbo Luan. 2015. Joint learning of character and word embeddings. In *Proceedings of IJCAI*, pages 1236–1242.

K Bretonnel Cohen, Karin Verspoor, Helen L Johnson, Chris Roeder, Philip V Ogren, William A Baumgartner Jr, Elizabeth White, Hannah Tipney, and Lawrence Hunter. 2009. High-precision biological event extraction with a concept recognizer. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing: Shared Task*, pages 50–58. Association for Computational Linguistics.

Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine Learning*, 20(3):273—297.

Jennifer D'Souza and Vincent Ng. 2012. Anaphora resolution in biomedical literature: A hybrid approach. In *Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, pages 113–122. ACM.

Kai Hakala, Sofie Van Landeghem, Tapio Salakoski, Yves Van de Peer, and Filip Ginter. 2013. Evex in st13: Application of a large-scale text mining resource to event extraction and network construction. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 26–34. Association for Computational Linguistics.

Halil Kilicoglu and Sabine Bergler. 2011. Adapting a general semantic interpretation approach to biological event extraction. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 173–182. Association for Computational Linguistics.

Jin-Dong Kim, Tomoko Ohta, Sampo Pyysalo, Yoshinobu Kano, and Jun'ichi Tsujii. 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*, pages 1–9. Association for Computational Linguistics.

Jin-Dong Kim, Yue Wang, Toshihisa Takagi, and Akinori Yonezawa. 2011. Overview of genia event task in bionlp shared task 2011. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 7–15. Association for Computational Linguistics.

Jin-Dong Kim, Yue Wang, and Yamamoto Yasunori. 2013. The genia event extraction shared task, 2013 edition-overview. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 8–15. Association for Computational Linguistics.

Chen Li, Maria Liakata, and Dietrich Rebholzschuhmann. 2013. Biological network extraction from scientific literature: State of the art and challenges. *Briefings in Bioinformatics*, 15(5):856–877.

Chen Li, Runqing Song, Maria Liakata, Andreas Vlachos, Stephanie Seneff, and Xiangrong Zhang. 2015a. Using word embedding for bio-event extraction. *ACL-IJCNLP 2015*, page 121.

Lishuang Li, Rui Guo, Zhenchao Jiang, and Degen Huang. 2015b. An approach to improve kernel-based protein–protein interaction extraction by learning from large-scale network data. *Methods*, 83:44–50.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Makoto Miwa, Rune Sætre, Yusuke Miyao, and Junichi Tsujii. 2009. Protein–protein interaction extraction by leveraging multiple kernels and parsers. *International journal of medical informatics*, 78(12):e39–e46.

Makoto Miwa, Paul Thompson, and Sophia Ananiadou. 2012. Boosting automatic event extraction from the literature using domain adaptation and coreference resolution. *Bioinformatics*, 28(13):1759–1765.

Yusuke Miyao, Kenji Sagae, Rune Sætre, Takuya Matsuzaki, and Jun'ichi Tsujii. 2009. Evaluating contributions of natural language parsers to protein–protein interaction extraction. *Bioinformatics*, 25(3):394–400.

Tsendsuren Munkhdalai, Oyun-Erdene Namsrai, and Keun H Ryu. 2015. Self-training in significance space of support vectors for imbalanced biomedical event data. *BMC bioinformatics*, 16(Suppl 7):S6.

Claire Nédellec, Robert Bossy, Jin Dong Kim, Jung Jae Kim, Tomoko Ohta, Sampo Pyysalo, and Pierre Zweigenbaum. 2013. Overview of bionlp shared task 2013. In *Bionlp Shared Task 2013 Workshop*, pages 1–7.

Nikolas Papanikolaou, Georgios A Pavlopoulos, Theodosios Theodosiou, and Ioannis Iliopoulos. 2015. Protein–protein interaction predictions using text mining methods. *Methods*, 74:47–53.

Yifan Peng, Samir Gupta, Cathy H Wu, and K Vijay-Shanker. 2015. An extended dependency graph for relation extraction in biomedical texts. *ACL-IJCNLP 2015*, page 21.

Vasin Punyakanok, Dan Roth, and Wen-tau Yih. 2008. The importance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics*, 34(2):257–287.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014. Learning sentiment-specific word embedding for twitter sentiment classification. In *ACL (1)*, pages 1555–1565.

Domonkos Tikk, Illés Solt, Philippe Thomas, and Ulf Leser. 2013. A detailed error analysis of 13 kernel methods for protein–protein interaction extraction. *BMC bioinformatics*, 14(1):1.

Huazheng Wang, Bin Gao, Jiang Bian, Fei Tian, and Tie-Yan Liu. 2015. Solving verbal comprehension questions in iq test by knowledge-powered word embedding. *arXiv preprint arXiv:1505.07909*.