PPI-IRO: a two-stage method for protein–protein interaction extraction based on interaction relation ontology

Chuan-Xi Li

Institute of Intelligent Machines,
Chinese Academy of Sciences,
Hefei 230031, China

and

School of Information Science and Technology,
University of Science and Technology of China,
Hefei 230026, China
E-mail: ben04@mail.ustc.edu.cn

Peng Chen*

Institute of Intelligent Machines,
Chinese Academy of Sciences,
Hefei 230031, China

and

School of Information Science and Technology,
University of Science and Technology of China,
Hefei 230026, China
Fax: 86-551-5592420
E-mail: pchen@iim.ac.cn

and

Mathematical and Computer Sciences and Engineering Division,
King Abdullah University of Science and Technology,
Thuwal 23955-6900, Kingdom of Saudi Arabia
*Corresponding author

Ru-Jing Wang

Institute of Intelligent Machines,
Chinese Academy of Sciences,
Hefei 230031, P.R. China

and

School of Information Science and Technology,
University of Science and Technology of China,
Hefei 230026, China
E-mail: rjwang@iim.ac.cn

Xiu-Jie Wang

State Key Laboratory of Plant Genomics,
Institute of Genetics and Developmental Biology,
Chinese Academy of Sciences,
Beijing 100101, China
E-mail: xjwang@genetics.ac.cn

Ya-Ru Su

Institute of Intelligent Machines,
Chinese Academy of Sciences,
Hefei 230031, China

and

School of Information Science and Technology,
University of Science and Technology of China,
Hefei 230026, China
E-mail: smomo@mail.ustc.edu.cn

Jinyan Li

Advanced Analytics Institute,
University of Technology Sydney,
Australia
E-mail: jinyan.li@uts.edu.au

Abstract: Mining Protein-Protein Interactions (PPIs) from the fast-growing biomedical literature resources has been proven as an effective approach for the identification of biological regulatory networks. This paper presents a novel method based on the idea of Interaction Relation Ontology (IRO), which specifies and organises words of various proteins interaction relationships. Our method is a two-stage PPI extraction method. At first, IRO is applied in a binary classifier to determine whether sentences contain a relation or not. Then, IRO is taken to guide PPI extraction by building sentence dependency parse tree. Comprehensive and quantitative evaluations and detailed analyses are used to demonstrate the significant performance of IRO on relation sentences classification and PPI extraction. Our PPI extraction method yielded a recall of around 80% and 90% and an F1 of around 54% and 66% on corpora of AIMed and BioInfer, respectively, which are superior to most existing extraction methods.

Keywords: protein–protein interaction; interaction relation ontology; relation word; sentence typed dependency; relation extraction; text mining; information extraction; bioinformatics.

Reference to this paper should be made as follows: Li, C-X., Chen, P., Wang, R-J., Wang, X-J., Su, Y-R. and Li, J. (2014) ‘PPI-IRO: a two-stage method for protein–protein interaction extraction based on interaction relation ontology’, *Int. J. Data Mining and Bioinformatics*, Vol. 10, No. 1, pp.98–119.

Biographical notes: Chuan-Xi Li received his Bachelor’s Degree in Computer Science and Technology from the University of Science and Technology of China. He is currently a PhD Candidate in Pattern Recognition and Intelligent System, University of Science and Technology of China. His research interests include text mining, bioinformatics and information extraction.

Peng Chen is currently an Associate Professor in the Institute of Intelligent Machines, Chinese Academy of Sciences. He received his Bachelor degree from Electronic Engineering Institute, his Master degree from Kunming University of Science and Technology and his PhD degree from University of Science and Technology of China. From January 2006 to June 2006, he was a Senior Research Associate in City University of Hong Kong. During April 2008–April 2009, he was a Postdoctoral Research Fellow in Howard University, USA. From July 2009 to December 2010, he worked in Nanyang Technological University, Singapore, as a Postdoctoral Research Fellow. His research interests include machine learning and data mining with applications to pattern recognition, bioinformatics, etc. He has published more than 20 journal/conference articles including: BMC Bioinformatics, Amino Acids, FEBS Letters, etc.

Ru-Jing Wang received his PhD in Pattern Recognition and Intelligent System from the University of Science and Technology of China. He currently a Professor of Pattern Recognition and Intelligent System at the University of Science and Technology of China and Institute of Intelligent Machines of Chinese Academy of Sciences. His research interests include data mining, semantic web, intelligent computing, agricultural ontology, intelligent decision and knowledge engineering.

Xiu-Jie Wang is currently an investigator in the Institute of Genetics and Developmental Biology, Chinese Academy of Sciences. She received his BSc (Biology) from Nankai University (China), her MSc (Biochemistry) from the Hong Kong University of Science and Technology (China) and her PhD (Bioinformatics) from the Rockefeller University (USA). Hers research interests include:

- computational discovery of non-coding RNA genes and their regulatory mechanisms
- identification of biological roles of non-coding RNAs in eukaryotic gene expression regulatory networks.

She has published more than 40 journal papers. These journals include: *Nature, J. Biol Chem.*, *Genomics*, *Genome Biol.*, *FEBS Lett.*, *Cell Res.*, *Plant Cell Physiol.*, *Plant Cell*, *Proc. Natl. Acad. Sci USA*, *Genes and Dev.*, *Nucleic Acids Res.*, *BMC Genomics*, *Plant J.*, *Front. Biol.*, etc.

Ya-Ru Su received her Bachelor's Degree in Automation from the University of Science and Technology of China; she is currently a PhD candidate in Pattern Recognition and Intelligent System, University of Science and Technology of China. Her research interests include data mining and bioinformatics.

Jinyan Li obtained his Bachelor's Degree of Science (Applied Mathematics) from National University of Defense Technology (China), his Master's Degree of Engineering (Computer Engineering) from Hebei University of Technology (China) and his PhD (Computer Science) from the University of Melbourne (Australia). He joined UTS in March of 2011 after ten years of fascinating research and teaching work in Singapore (Institute for Infocomm Research, Nanyang Technological University and National University of Singapore). His research interests include bioinformatics, computational biology, data mining, graph theory, information theory, machine learning and theoretical biology. He has published 51 journal papers and 60 conference papers, of which 24 journal papers and 32 conference papers are in the ERA ratings of A/A*.

1 Introduction

Gene regulation and protein interaction play fundamental roles in controlling complex biological processes. Mining PPIs (PPIs (De Las Rivas and Fontanillo, 2010) are understood as physical contacts with molecular docking between proteins that occur in a cell or in a living organism in vivo.) It can reduce the tedious manual effort of biologists to read the large amount of literature texts from biomedical literature resources and it also has the potential to improve the connections between the annotations in biological databases and the supporting evidence contained in the relevant literature documents. Moreover, extracted PPIs can serve as a complementary knowledge to the existing biological databases, such as DIP, BIND and INTACT (Xenarios et al., 2002; Bader et al., 2001; Aranda et al., 2009). To meet the ever-growing demands, some research projects have been launched in the communities of NLP and text mining, such as BioNLP, TREC and BioCreAtIvE II, II.5 and III (Kim et al., 2009; Cohen and Hersh, 2006; Krallinger et al., 2008; Leitner et al., 2010). Some excellent online tools were also developed. For example, iHop (Hoffmann and Valencia, 2005) considered proteins and genes as hyperlinks between sentences and it used abstracts to navigate gene networks and further to facilitate the exploration of protein interaction processes. PLAN2L (Krallinger et al., 2009) provided an online text-mining tool to extract information related to Arabidopsis. SEBINI (Taylor et al., 2009) created a structured workflow for protein–protein network inference and supplemental analysis.

Many machine learning methods have also been proposed for PPI extraction. A comprehensive comparison study on various kernel-based methods on PPI extraction was conducted and total 19 methods were evaluated on five common corpora (Tikk et al., 2010). Those kernel-based methods, such as convolution kernels (Collins and Duffy, 2001), effectively identified PPIs using a deep syntactic parser of sentences. A novel domain-based kernel method was proposed to predict PPIs by Chen et al. (2008). The PIE system (Kim et al., 2008) utilised grammatical structures to filter PPI sentences and then adopted SVM with a convolution tree kernel to extract PPIs from the GENIA corpus, where the sentences parse tree and the interaction word dictionary were combined to filter the PPI sentences. Although machine learning methods have produced promising results on PPI extraction, their performances varied greatly on different corpora in real-world applications.

Relation words are important feature indicators of protein interaction relations. Observed by iHop (Hoffmann and Valencia, 2005), about 90% of active relationships of proteins can be expressed syntactically as ‘protein verb protein’, highlighting the importance of interaction verbs at online relation navigation networks. Interestingly, all of the 53 frequent verbs can be used to model gene-verb-gene patterns as well. Recently, BioPPIExtractor (Yang et al., 2009) considered interaction words recognition as an important task prior to grammar parsing through a use of 154 verbs and their variants. MedScan (Novichkova et al., 2003) applied a context-free grammar and lexicon to process sentences and a series of regularised logical structures were constructed to understand sentence semantics; the generated semantic structure of sentences also utilised predicate-argument relations between protein entities. As described by Saetre et al. (2008), predicate features can improve PPI extraction performance, while our idea using relation words of IRO makes use of both the predicate features and the modifying features. The work of Rebholz-Schuhmann et al. (2010) provided a survey on the relation verbs used by different research teams and described the prediction capacity of different verbs for PPI extraction on the existing corpus.

Lexical and syntactic analyses are widely used in PPI extraction to capture the dependency relation of sentence elements. The importance of syntactic features in PPI extraction was evaluated on five benchmark data sets in Fayruzov et al. (2008), which concluded that deep syntactic information can achieve a good performance. Actually, deep syntactic parser and shallow dependency parser were both used to capture the semantic meaning of sentences (Saetre et al., 2008) and rules were created to discover protein interactions by SVM with tree kernels, taking Head-Driven Phrase Structure Grammar (HPSG), a highly lexicalised and non-derivational generative grammar theory developed by Pollard and Sag (1994), as dependency parser. IntEx (Ahmed et al., 2005) divided complex sentences into simple clausal structures and extracted interaction relations by analysing the syntactic roles and linguistically significant combinations. Work in Liu et al. (2010) applied sentence dependency tree to extract PPIs and it can provide detailed comparison of different feature effects on common corpus. One more example is about RelEx (Fundel et al., 2007), which extracted relations based on an NLP technique producing dependency parse trees constrained by three simple rules.

This paper presents a two-stage PPI extraction method, which consists of a relation sentence classification and a PPI extraction. Our IRO has three levels and stores a total of 1383 words. It is used in relation sentence classification for assessing the importance of interaction relation words between proteins and is also used in PPI extraction through a sentence dependency parse tree.

In the first stage, we employ a binary classifier to identify relation sentences and non-relation sentences instead of using multiple protein names co-occurrence to filter out non-relation sentences, as the method taking by Polajnar et al. (2011). The weights of the words in IRO can be increased for highlighting its importance during the feature selection. In the second stage, PPI extraction, we construct a *relation dependency forest* for each relation sentence based on typed dependencies parse tree from the Stanford parser (De Marneffe et al., 2006). For a relation dependency forest, the total number of its trees is the amount of relation words in the sentence. The relation dependency trees are constructed according to typed dependencies. The root node of the tree is a relation word occurring in the sentence and the internal nodes and leaf nodes of the tree are the syntactic-related words with root node in the typed dependencies. The weight of the tree is assigned according to the weight of the relation words of IRO and the weights of the protein pairs are then extracted from the tree. The weight of the pair reflects the confidence level of the extracted PPIs. The final extracted PPIs are confirmed by using a combination of two parameters: one controlling the influence of the number of the trees and one controlling the weight of the protein pair.

In our method, the PPI extraction process is corpus independent, whereas the traditional learning-based PPI extraction methods, Bayesian inference method (Chowdhary et al., 2009), maximum entropy model (Sun et al., 2007) and those kernel methods proposed in Zelenko et al. (2003), Bunescu and Mooney (2006) and Miwa et al. (2009) are not. Moreover, our method has a low computational complexity. Compared with a simple rule-based method, our method can cover the protein interaction pairs in sentences more comprehensively and does not depend on specific rules but only the syntactic dependency relation between proteins.

2 Methods

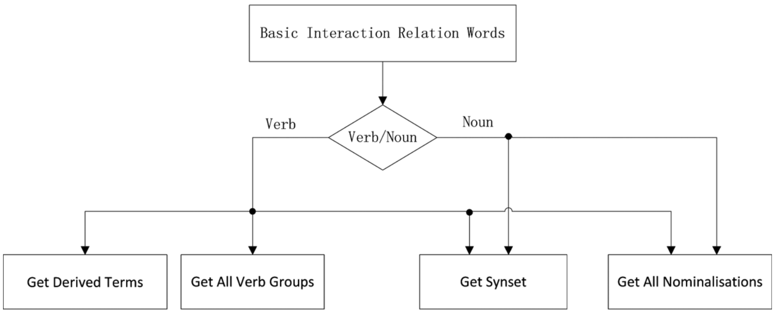
In this section, we describe our method in details. The construction process of IRO is presented first. Then, the framework of our method is followed, which consists of two main components: relation sentences classification and PPI extraction.

2.1 Construction of Interaction Relation Ontology (IRO)

In general, ontology consists of entities and their relations, whereas entities contain different description attributes. Specifically for lexicon, its structure is flat and just contains the words without relations and attributes. In IRO, there are two inherent relations between interaction relation words, defined as *has-extended* and *is-a*. The interaction relation words have the Part-of-Speech (POS) attribute. During the core relation words extension, different core relation words are allowed to have the same extended child words, which are not allowed in lexicon but which is in accordance with the definition of ontology.

Initially, a set of 154 core relation words was manually selected from the BioInfer (Pyysalo et al., 2007) corpus and pathway studio (Nikitin et al., 2003). Then, PPIs containing more than three sentences from pathway studio were selected and biologists are requested to pick out the interaction relation words. For the BioInfer corpus, we used the annotated information to extract the interaction relation words and requested the biologist to validate. The final core relation words assemble the two subsets of words. The core relation words were then extended by WordNet (Miller, 1995). If the POS of a core relation word is noun, the forms of nominalisation and synset are taken for further use; if the POS is verb, the forms of verb groups and derived terms are also considered. In this work, we use the POS as an attribute of the relation words. The whole procedure is shown in Figure 1.

Figure 1 The IRO construction by words extension



A word is considered as an available extended word, if its frequency is larger than threshold f_t, where f_t is set to 2. It is also set to other values in this work, such as 3 and 4, however, we found that in such cases it leads to dramatically decreased amount of extended words. A pseudocode of computational steps for the construction of IRO is shown as Algorithm 1.

Algorithm 1 Interaction Relation Ontology Construction

Input: core relation words *CR*; WordNet JAVA Interface *RitaWN* (*Howe, 2010*); threshold f_t
Output: Interaction Relation Ontology *IRO*
Function IRO Construction (*CR*, *RitaWN*)

```
For verb in CR
    Get its extended verb groups by RitaWN and store them into FreqExtendedVerbs if
    the frequencies of extended verbs are larger than threshold $f_{t;}$
Interaction relation ontology IRO = empty;
IRO.root = RelationWordEntity;
For word in core relation words
    word.addAttribute (POS);
    IRO.root.child = word;
    Add the Synset (RitaWN) as word children and set its attribute POS;
    Add the Normalisation (RitaWN) as word children and set its attribute POS;
    If the word is verb
        Add the Verbs (WordNet) as word children and set its attribute POS, if the verb
        is in FreqExtendedVerbs;
    Add the DerivedTerms (RitaWN) as word children and set its attribute POS;
Return IRO
```

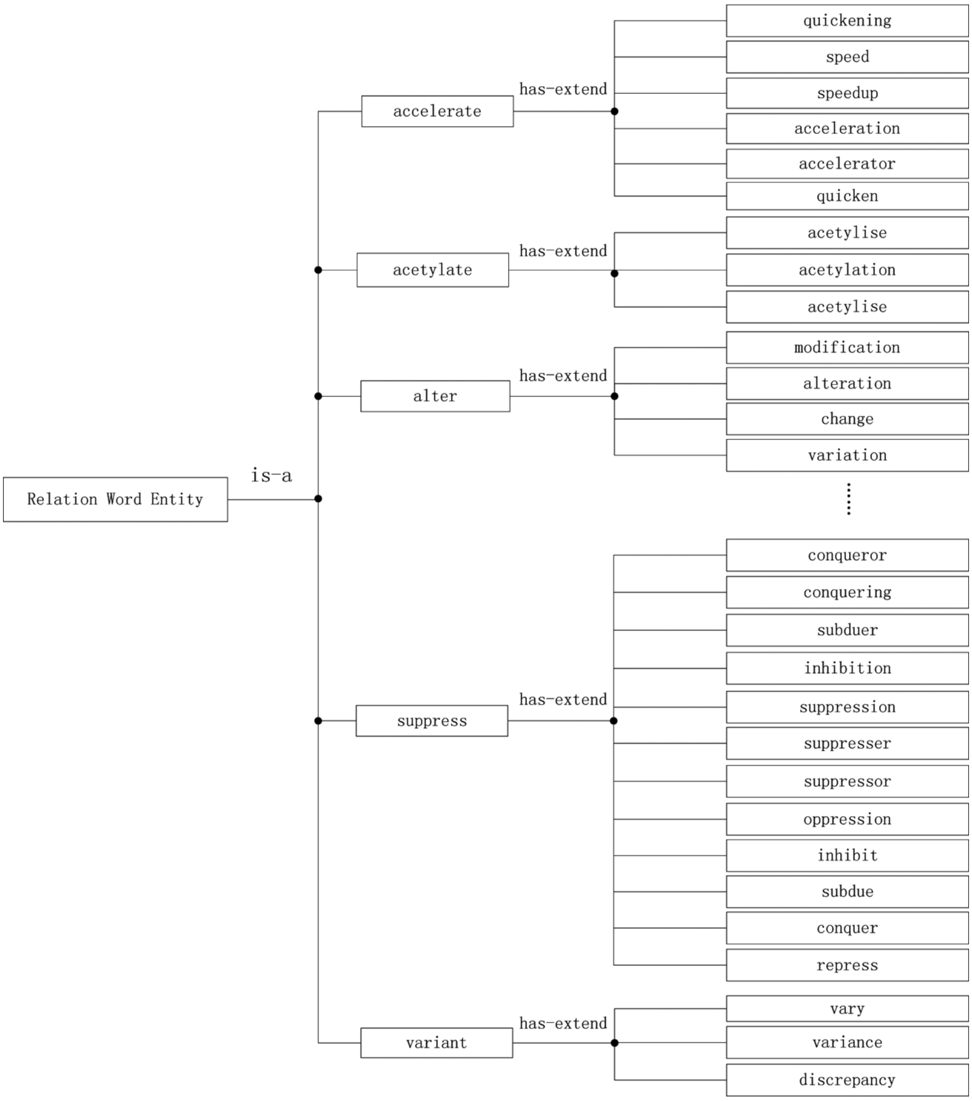
The constructed IRO has three levels and contains 1383 words, as illustrated partially in Figure 2. In the feature selection of relation sentence classification as well as the dependency tree filtering and PPI extraction, the core relation words have larger weights than those of the extended words.

2.2 *Framework of our proposed method*

As mentioned earlier, our method consists of two stages of processes: interaction relation sentences classification and PPI extraction. The first stage identifies PPI relation sentences and non-PPI relation sentences and the second stage extracts protein interaction pairs from those interaction relation sentences produced by the first stage. Figure 3 shows a detailed diagram of these processes.

To avoid similar sentences in the classification, sentence normalisation is performed in the first stage, which removes repeated sentences and those sentences that just differ in punctuations. During feature extraction and selection, the weights of relation words in the

Figure 2 Interaction Relation Ontology (IRO). Relation word entity is the root of the ontology, the middle layer is the core relation words and the rightmost layer is the extended words

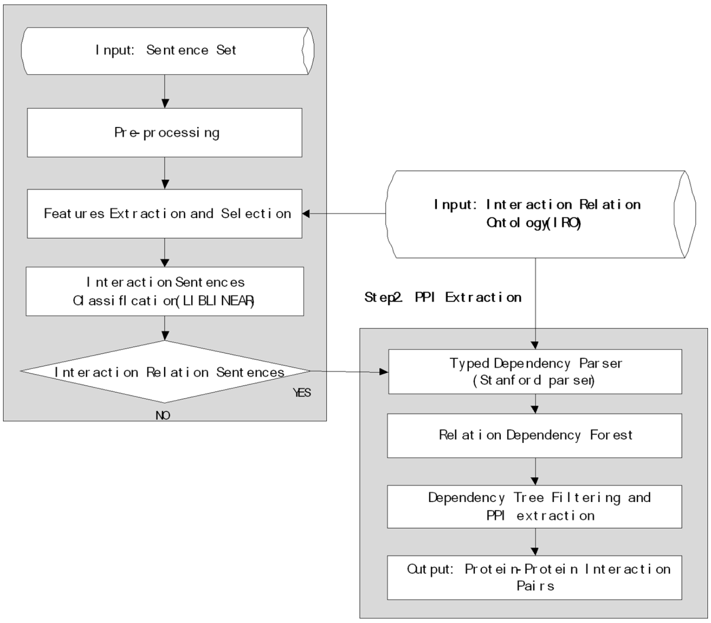


sentences, such as important features BOW, bi-gram and tri-gram, are increased manually to improve the classification performance. At the second stage (PPI extraction), for each sentence, we construct a relation dependency forest that consists of several dependency trees. The dependency trees are constructed according to the typed dependencies of the sentence. Finally, dependency tree filtering and PPI extraction are conducted to extract protein pairs from the relation dependency forest.

2.2.1 Classification for protein interaction relation sentences

Classification of PPI sentences includes three steps: sentence pre-processing, features extraction and selection and sentence classification.

Figure 3 Framework of the proposed method


Input: Sentence Set
Pre-processing
Features Extraction and Selection
Interaction Sentences Classification(LIBLINEAR)
Interaction Relation Sentences
YES
NO
Input: Interaction Relation Ontology(IRO)
Step2. PPI Extraction
Typed Dependency Parser (Stanford parser)
Relation Dependency Forest
Dependency Tree Filtering and PPI extraction
Output: Protein-Protein Interaction Pairs

Sentence pre-processing: This step is to eliminate the effect of protein names on sentence dependency parse by performing name normalisating, bracket cleaning and references and replicated sentences removing. In name normalisation, all of the protein names are mapped to a standard form 'ProteinXXXXX', namely the prefix 'Protein' plus a digital number with 5 bits. For instance, the form 'Arp2/3' in the case 'Arp2/3 complex from Acanthamoeba binds profilin and cross-links actin filaments', it is normalised to Arp2 and Arp3 and then they are mapped to two standard forms *Protein00314* and *Protein00315*. Bracket cleaning removes the contents in the bracket, such as in the sentence " … including actin (twofold to threefold) … increases in cellular protein content (20–40%)", whatever in the bracket is removed. References in a sentence will also be removed in sentence pre-processing. Here, Levenshtein Distance (Gusfield, 2007), which is also used by Rudniy et al. (2010), is adopted to eliminate the repetition or very similar sentences only with different punctuations, case matters, etc. Moreover, the annotated protein names of corpora are mapped to annotate entity names without considering un-annotated protein name. A pseudocode of the sentence pre-processing step is described as Algorithm 2, where parameter d_t is used to measure the similarity of two sentences. Two sentences are assumed to have the same contents if the distance between the two sentences is shorter than d_t.

Algorithm 2 Sentence pre-processing

```
Input: corpus
Output: Normalised Sentence Set
1. Retrieve the protein names list from corpus and store in ProteinNameSet;
2. Normalise the protein name in ProteinNameSet, including replace all the comma, dot,
   slash, backslash with underline;
```

3. Map all of protein names of *ProteinNameSet* into standard form '*ProteinXXXXX*';
4. For sentence in sentences;
5. Clear the irrelevant contents of sentence, including references, the contents in bracket, etc.;
6. Remove the similar sentence if the Levenshtein Distance of two sentences is less than threshold $d_{t;}$
7. Replace the protein names in sentence with a standard form '*ProteinXXXXX*' according to *ProteinNameSet*;
8. Return normalised sentences

Features extraction and selection: This step produces a Vector Space Model (VSM) of the sentences. We choose BOW (bag of word), POS, bigram and trigram as features of sentences. The weights of relation words in IRO are boosted manually.

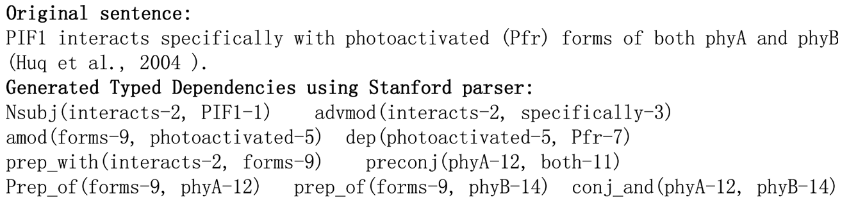
Sentences classification: In this work, Naïve Bayes, Bagging, J48 and SMO from WEKA (Hall et al., 2009) are applied to evaluate the effects of different feature combinations on classification performance and we choose LIBLINEAR (Fan et al., 2008) as our sentence classifier. The bigram and trigram features containing relation words in IRO are re-weighted manually to highlight the importance of interaction relation words between proteins in sentence classification. The core relation word features of IRO have higher weights than those of extended relation word features and are set to 5 and 3 times larger than those of the general features, respectively.

2.2.2 PPI extraction

The relation sentences produced in *protein interaction relation sentences classification* are used to extract protein interaction pairs in *PPI extraction*. PPI extraction consists of three sub-steps: sentence dependency parse, relation dependency forest construction and dependency tree filtering and PPI extraction.

Sentence dependency parse: Stanford parser is adopted to parse relation sentences and generates the typed dependencies of the sentence. Figure 4 shows an example of an original sentence and its typed dependencies.

Figure 4 Example sentence and its typed dependency



```
Original sentence:
PIF1 interacts specifically with photoactivated (Pfr) forms of both phyA and phyB
(Huq et al., 2004 ).
Generated Typed Dependencies using Stanford parser:
Nsubj(interacts-2, PIF1-1)    advmod(interacts-2, specifically-3)
amod(forms-9, photoactivated-5)  dep(photoactivated-5, Pfr-7)
prep_with(interacts-2, forms-9)    preconj(phyA-12, both-11)
Prep_of(forms-9, phyA-12)   prep_of(forms-9, phyB-14)  conj_and(phyA-12, phyB-14)
```

Relation dependency forest construction: On the basis of the typed dependencies of a sentence and IRO, a relation dependency forest of the relation sentence is constructed by assembling several dependency trees. The root node of the tree is the relation word and the internal and leaf nodes are the words that have syntactical dependency in the typed dependencies of the sentence with the same root node. The paths from the root to a leaf node depend on their dependency relations in the typed dependencies. The complete procedure is described in Algorithm 3. In Algorithm 3, line 1 defines the protein pairs in the coordinating relation of the typed dependency as *non-protein interaction pairs*, which are *noun compound modifier, appositional modifier, coordination modifier, possession modifier, conjunction and, conjunction or, or abbreviation,* between a governor and a dependent. Here, we do not consider the case that any other potential interaction pair exists in the coordinating relation. Details of typed dependencies representation can be found in de Marnee and Manning (2010).

Algorithm 3 Relation dependency forest construct

Input: interaction relation ontology *IRO*, typed dependencies
Output: sentence dependency forest *DF*,

```
1. Definition of coordination relation according to typed dependencies, Coordinating
      Relation = “nn, appos, cc, ppos, conj_and, conj_or, abbrev”;
2. Get the proteins pairs of CoordinatingRelation from typed dependencies of sentence
      (TypedDependencies) and annotate them as non-protein-pairs;
3. Remove the typed dependencies pairs of coordination relation from TypedDependencies;
4. for all the interaction relation words (irw) in interaction relation ontology
5. Use the irw as root to build the relation dependency tree, where the inner nodes and
   leaf nodes are formulated according to the pairs in TypedDependencies and all the trees
   constitute sentence dependency forest;
6. return sentence dependency forest
```

Dependency Tree filtering and PPI extraction: This sub-step filters sentence dependency tree and generates protein interaction pairs. The procedures of generating PPI pairs by dependency tree filtering and PPI extraction are shown in Algorithm 4. Initially, only protein names and relation words in a relation dependency tree are retained. After removing irrelevant nodes, the dependency tree of sentence is classified into six types, which are shown in Figure 5.

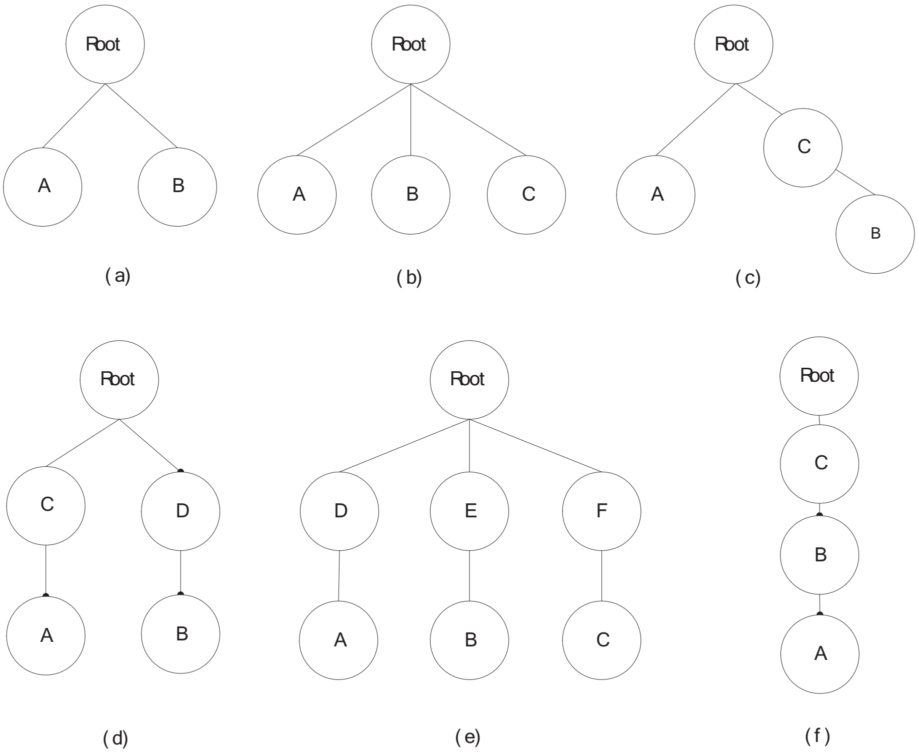
A protein pair is considered as an interaction pair if its frequency is larger than the threshold f_{pair}*DF.size*, where *DF.size* is the number of dependency trees in a dependency forest. The parameter f_{pair} is used to measure the influence of the number of the relation words and the proteins pair occurred in most of the dependency trees is considered as a potential PPI. The parameter ω is used to control the weight of the proteins pair, which is set according to the weight of an interaction relation word.

Algorithm 4 Dependency tree filtering and PPI extraction

Input: Dependency Forest *DF*, Protein Name Set *PNS*, threshold f_{pair}, the weight *W* of the root node
Output: protein-protein interaction *PPI*

```
1. Remove the non-protein name from the tree of the DF;
2. $V_C$ = empty; $V_{CP}$ = empty; PPI = empty;
3. For all the tree in forest
4.       Get the weight of relation word, $\omega$
```

Figure 5 Six types of dependency tree. The root node root is a relation word, where the capital one-letter represents protein name



```
    For each sub-tree $Si$ of the tree
        $V_{Ci} = \omega$*Combination ($Si$);
        $V_C = V_C \cup V_{Ci;}$
    For any sub-tree $Si,Sj$ ($i \neq j$) of the tree
        $V_{CPij} = \omega$*CartesianProduct ($Si,Sj$);
        $V_{Cp} = V_{Cp} \cup V_{Cpij;}$
V= $V_C \cup V_{CP;}$
For all pairs in $V$
    If count (pair) > forest. size*$f_{\text{pair}}$
        $PPI$.add (pair);
Return $PPI$
```

Let (A, B) be a symmetric and transfer relation between proteins A and B. For one sentence S, we treat each sub-tree of the root node as a set S_i (i is the number of the sub-trees of the root). The total extracted protein pairs of the tree consist of two parts. One part is the inner-sub-tree pairs V_C. Let V_{Ci} represent the combination of the elements in S_i, where $V_{Ci} = (S_{ij}, S_{ik})$, $j \neq k$, $S_{ij} \in S_i$, $S_{ik} \in S_i$, the size of V_{Ci} is $C^2|s_i|$ and the total inner-sub-tree pairs of one tree are $V_C = \cup V_{Ci}$. The other is inter-sub-tree pairs V_{CP}. For all sets S_i and S_j, where $S_i \neq S_j$, we make their Cartesian product, denoted by V_{CPij}, where $V_{CPij} = S_i \times S_j = (S_{ij}, S_{jk})$, $i \neq j$, $S_{ij} \in S_i$, $S_{ik} \in S_i$. Therefore, the total inter-sub-tree pairs of a tree are $V_{CP} = \cup V_{CPij}$,($i \neq j$). The final protein interaction pairs of one sentence consist of V_{CP} and V_C, i.e., $V = V_C \cup V_{CP}$.

Taking the sentence in Figure 4 as an example, its dependency forest is illustrated in Figure 6 and the extracted results are presented in Figure 7.

Note that a larger frequency of a PPI pair reflects a higher reliability of extraction of protein interaction relations.

Figure 6 Sentence dependency forest. The root node is relation word and the other nodes are protein names

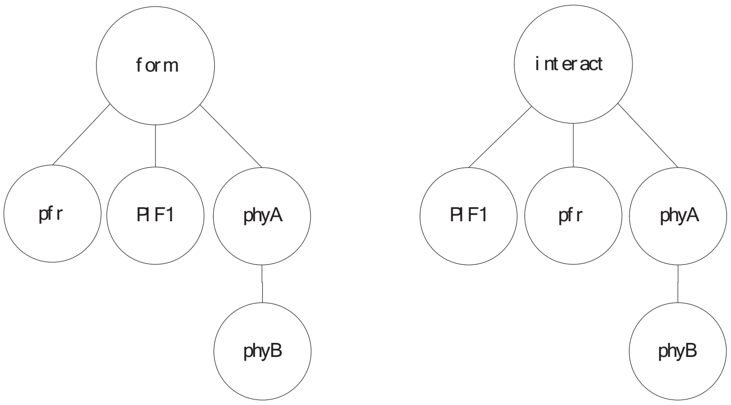
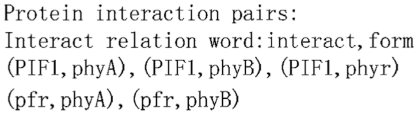


Figure 7 Dependency tree filtering and PPI extraction results


```
Protein interaction pairs:
Interact relation word:interact,form
(PIF1,phyA),(PIF1,phyB),(PIF1,phyr)
(pfr,phyA),(pfr,phyB)
```

3 Experimental results and analysis

Our PPI extraction method was evaluated on corpora of BioInfer (Pyysalo et al., 2007) and AIMed (Bunescu et al., 2005), which contain 849 and 238 Medline abstracts, respectively. In total, 7023 protein references and about 2908 annotated interactions are involved in these two data sets.

RitaWN (Howe, 2010) interface tools were used to invoke WordNet ontology. The parameter d_t used to remove similar sentences (by Algorithm 2) was set as 4; the parameter f_{pair} in the dependency tree filtering and PPI extraction (by Algorithm 4) was set as 0.3 for filtering out proteins pairs with a low frequency.

3.1 Results of classification on relation sentences and non-relation sentences

Assume that IRW is an interaction relation word in IRO. Features selected in the first stage (relation sentence classification) contain BOW, POS, bigram, trigram and IRW according to Van Landeghem et al. (2010). The weights of features with higher level than IRW are increased 5 times and those at lower levels 3 times. The values of the core relation words and the extended relation words are set manually and we found that the best classification results are achieved for values 5 and 3, respectively. Before the classification, a resample technique is used on the training samples with parameter ‘*Resample -B 0.0 -S 1 -Z 100.0*’ in WEKA (Hall et al., 2009).

Four measurements, precision, recall, F1 score and AUC (area under the receiver operating characteristic curve), are used to evaluate the performance of our method. The runtime costs of all classifiers are presented in column *RT* in Table 1. All results are obtained by a 10-fold cross-validation strategy. Among classifiers, J48, Bagging, SMO and LIBLINEAR, LIBLINEAR is taken as our relation sentence classifier owing to its less runtime cost. Here, the cost of SMO classifier is around 45 times bigger than LIBLINEAR. The parameters of the classifiers of J48, Bagging, SMO and LIBLINEAR are set as ‘*-C 0.25 -M 2*’, “ *-P 100 -S 1 -num-slots 1 -I 10 -W REPTree – -M 2 -V 0.0010 -N 3 -S 1 -L -1 -I 0.0*”, “ *-C 1.0 -L 0.0010 -P 1.0E-12 -N 0 -V -1 -W 1 –K*”, “*supportVector.PolyKernel -C 250007 -E 1.0*” and “ *-S 1 -C 1.0 -E 0.01 -B 1.0*”, respectively.

For comparison, we present all of the classification results in Table 1.

Table 1 Classification results on relation sentences and non-relation sentences. The column RT(s) represents the runtime and unit is second

	AIMed					*BioInfer*				
	Precision	*Recall*	*F score*	*AUC*	*RT*(s)	*Precision*	*Recall*	*F score*	*AUC*	*RT*(s)
NaiveBayes	0.814	0.796	0.801	0.842	2.25	0.825	0.826	0.826	0.879	0.81
J48	0.887	0.887	0.887	0.895	19.64	0.866	0.869	0.867	0.877	7.19
Bagging	0.892	0.892	0.892	0.915	43.42	0.867	0.87	0.864	0.904	15.3
SMO	0.921	0.917	0.919	0.904	39.28	0.892	0.894	0.893	0.856	13.11
LIBLINEAR	0.931	0.927	0.929	0.914	0.89	0.884	0.906	0.895	0.897	0.3
LIBLINEAR+IRW	**0.937**	**0.938**	**0.937**	**0.922**	**0.89**	**0.899**	**0.917**	**0.908**	**0.901**	**0.3**

The LIBLINEAR classifier (with or without IRW) improves almost all of the measurements (precision, recall, F1 score and AUC) significantly in comparison with other classifiers. Our results also show that the indicated words, such as ‘regulate’, ‘bind’ and ‘phosphorylate’, of protein interactions are important when distinguishing relation sentences from non-relation sentences. On the large-scale corpus AIMed, the performance improvement by LIBLINEAR with IRW is higher than that on the corpus BioInfer. Moreover, these experimental results do validate the significance of IRO on differentiating relation sentences from non-relation sentences.

3.2 PPI extraction results

Table 2 compares the performance of our method with other closely related methods on AIMed and BioInfer. Results obtained by Tikk et al. (2010), Fundel et al. (2007) and Liu et al. (2010) are also included in Table 2 for more comparison. The symbol ‘-’ represents that the corresponding performance could not be found in the corresponding paper. In the row ‘kernel’, CV, CL and cc represent cross-validation, cross-learning and cross-corpus, respectively.

Table 2 PPI extraction results compared with other methods on corpus AIMed and BioInfer. All the results of kernel-based methods are cited from Tikk et al. (2010)

	AIMed			*BioInfer*		
	Precision	*Recall*	*F1 score*	*Precision*	*Recall*	*F1 score*
Tikk et al. (2010)	CV/CL/CC	CV/CL/CC	CV/CL/CC	CV/CL/CC	CV/CL/CC	CV/CL/CC
SL	47.5/28.3/66.8	65.5/86.6/29.2	54.5/42.6/41.5	55.1/62.8/55.1	66.5/36.5/66.5	60.0/46.2/40.6
SpT	33.0/20.3/48.4	25.5/48.4/16.3	27.3/28.6/34.7	44./ 38.9/44.0	68.2/48.0/68.2	53.4/43.0/24.3
KBSPS	50.1/28.6/71.6	41.4/68.0/15.0	44.6/40.3/40.3	–/62.2/49.9	–/38.5/61.8	55.1/47.6/24.8
Edit	77.5/26.8/86.4	43.5/59.7/28.8	39.0/37.0/39.6	-/53.0/50.4	–/22.7/39.2	43.8/31.7/15.9
APG	52.9/30.5/56.5	61.8/77.5/14.0	56.2/43.8/37.9	56.7/58.1/60.2	67.2/29.4/61.3	60.7/39.1/22.5
RelEx (Fundel et al., 2007)	0.40	0.50	0.44	0.39	0.45	0.41
Liu et al. (2010)	0.634	0.488	0.547	–	–	0.598
PPI-IRO	**0.404**	**0.803**	**0.538**	**0.520**	**0.898**	**0.658**

On the BioInfer data set, our method outperforms other methods in F1 score and recall, especially in the recall measure. For the AIMed corpus, our method is also competitive to other methods on precision, recall and F1 score. One reason that our method achieves a higher recall than most of the others is because the indicated words of PPI relation can be covered by IRO mostly and that IRWs used as the root node of dependency tree to extract protein interaction pairs can cover protein interaction pairs comprehensively. In comparison with the rule-based method RelEx, we found that although our method achieves only a little improvement of precision over that of RelEx, the recall of our method is far higher than RelEx. On the one hand, the fixed set of 3 templates applied on dependency parse tree in RelEx are insufficient very much when dealing with the various writing styles, while the typed dependencies of sentences are not limited to the specific rules and therefore they can cover the protein relations comprehensively. On the other hand, the interaction relation words are from the constructed IRO, capturing PPIs better than the *relations* of RelEx do. Therefore, as long as the relation words, which contain our IRO and the syntactical dependency relations, existing between proteins are correctly parsed by the typed dependencies of sentences, the proteins pair can be always extracted.

Although some kernel-based methods perform better than ours under some metrics, our method is still attractive owing to the linear computational complexity. First, the runtime cost of our method is linear with the number of sentences, where the construction of the sentence dependency parse tree takes the most runtime. Along with the performance improvement of sentence dependency parse, the runtime cost of our method will decrease further. Second, our method is scalable to the large amount of biomedical literature resources. Third, our method achieves a higher recall than most of the others, owing to the wide coverage of IRO on the indicating words of protein interaction relation.

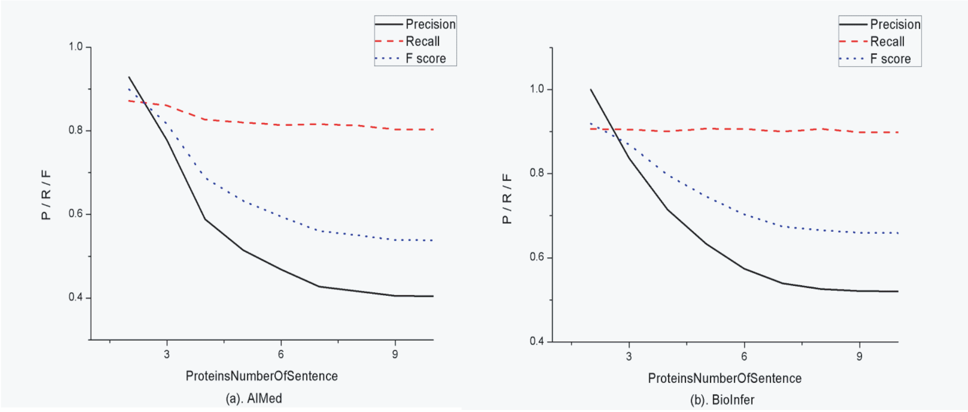
Parameter d_t in Algorithm 2 is used to measure the similarity between two sentences. If the distance between two sentences is shorter than d_t, the two sentences are assumed to describe the same content. We tested the parameters from 4 to 9 and could not find much difference in result and the parameter is then set to 4 in the experiments.

In Algorithm 4, parameter f_{pair} is used to measure the influence of the number of relation words; one protein pair highly occurred in the dependency trees is considered as a potential PPI. We set it to 0.3, namely we consider the protein pair as a possible PPI, if the ratio of the number of protein pairs to that of dependency trees exceeds 0.3. The other parameter ω is used to control the weight of the proteins pair, which is set depending on the weight of the dependency tree. The weight of the dependency tree is set according to the interaction relation word of the root node. We assigned each proteins pair a weight ω instead of just simply accumulating its number of different sentence dependency trees. The parameter ω is set as 1 or 0 according to the root node of protein pair from a core relation word or an extended relation word. The weight of extended relation words reflects the importance with respect to core relation words.

We also observed that the parameter f_{pair} can be set between 0.3 and 0.4, which did not have significant influence on experimental results. For the core relation words, the parameter ω is fixed as 1; for the extended relation words, ω can be set between 0.6 and 0.75, where we take the lower bound value. When the value of ω is larger than 0.75, the F1 score of PPI extraction was decreased. Moreover, if the numbers of proteins and relation words of the sentence are also small (less than 4 and 3, respectively), the values of f_{pair} and ω varying from 0 to 1 had no significant influence on the PPI extraction results only if ω is not less than f_{pair}.

We found that the accuracy of dependency parses becomes low and the precision of PPI extraction decreases accordingly, if sentences contain complicated structures or special letters, such as comma, period, colon and quotation. Moreover, as illustrated in Figure 8, the smaller the number of proteins in a sentence, the more effective our method is. With an increasing number of protein names in a sentence, the precision and recall of PPI extraction are decreased, whereas recalls appear no remarkable change. The large amount of proteins in a sentence leads to complicated dependency trees of the sentence and then worsens the performance of PPI extraction. Usually when the number of proteins is smaller than 7, our method turns to be more effective.

Figure 8 The impact of different protein numbers on PPI extraction performance. The P/R/F of vertical coordinate represent precision, recall and F1 score, respectively. (see online version for colours)



(a). AIMed

(b). BioInfer

Another factor influencing the performance of PPI extraction is the cross-name of proteins. For example, in the sentence ‘*Arp2/3 complex from Acanthamoeba binds profilin and cross-links actin filaments*’, the protein Arp3 was left out when performing protein name recognition, leading to all the relations related to Arp3 missing. Therefore, the effective protein name reorganisation can improve the performance of PPI extraction significantly.

We also compared the PPI extraction results on relation sentences from the classified results of Step 1 (SSS) with those relation Sentences Selected Manually (SSM). The experimental results in Table 3 show that all the ratio values of SSS/SSM are proportional to those of *LIBLINEAR* + IRW on precision, recall and F1 score. However, the values of SSS/SSM are bigger than those of the classified results. The reason lies in that although some non-relation sentences are misclassified as relation sentences, they cannot produce protein pairs in PPI extraction.

For example, although the sentence “*MSH2 plays a fundamental role in mispair recognition whereas MSH3 and MSH6* appear *to modify the specificity of this recognition*” in BioInfer corpus is classified as a relation sentence, no related protein pairs are extracted by our method. The typed dependencies of the sentence are shown in Figure 9. The Coordinating Relation is removed during the construction of its sentence dependency forest, which means that ‘*conj_and (MSH6, MSH3)*’ contains a non-relation interaction protein pair (*MSH6, MSH3*). Although the dependency forest is generated (see Figure 10), the pair

(*MSH6, MSH3*) is not considered as an interaction relation by PPI extraction and thus it is filtered out.

Table 3 Comparison of PPI-IRO extraction results based on relation sentences of Step 1 (binary classified results) and selected manually.

	AIMed			*BioInfer*		
	Precision	*Recall*	*F1 score*	*Precision*	*Recall*	*F1 score*
Sentences Selected from Step 1 (SSS)	0.404	0.803	0.538	0.520	0.898	0.658
Sentences Selected Manually (SSM)	0.422	0.829	0.559	0.543	0.909	0.680
The ratio of SSS and SSM	0.958	0.968	0.962	0.956	0.988	0.968
Sentences classified results of LIBLINEAR +IRW	0.937	0.938	0.937	0.899	0.917	0.908

Figure 9 Sentence typed dependency of sentence "MSH2 plays a fundamental role in mispair recognition whereas MSH3 and MSH6 appear to modify the specificity of this recognition"

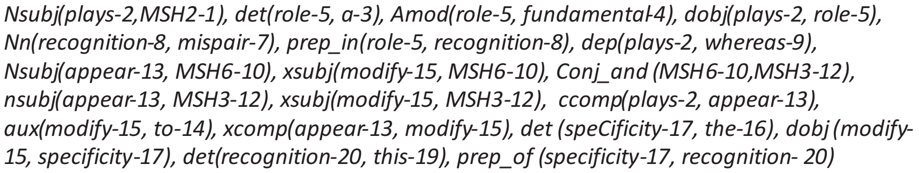
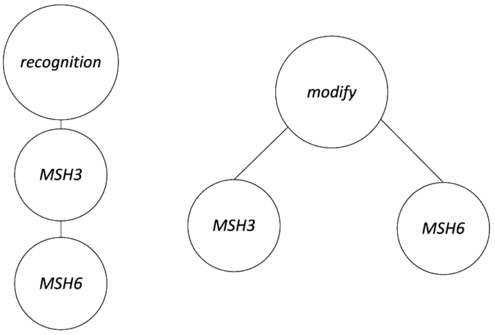

Nsubj(plays-2,MSH2-1), det(role-5, a-3), Amod(role-5, fundamental-4), dobj(plays-2, role-5), Nn(recognition-8, mispair-7), prep_in(role-5, recognition-8), dep(plays-2, whereas-9), Nsubj(appear-13, MSH6-10), xsubj(modify-15, MSH6-10), Conj_and (MSH6-10,MSH3-12), nsubj(appear-13, MSH3-12), xsubj(modify-15, MSH3-12), ccomp(plays-2, appear-13), aux(modify-15, to-14), xcomp(appear-13, modify-15), det (speCificity-17, the-16), dobj (modify-15, specificity-17), det(recognition-20, this-19), prep_of (specificity-17, recognition- 20)

Figure 10 Sentence dependency forest of Figure 9



4 Discussions

4.1 Related works

A survey conducted by Rebholz-Schuhmann et al. (2010) presents relation verbs used by different research teams and described prediction capacity of the different verbs for PPI extraction on the benchmark corpus. There are several differences between our work and the literature works discussed by Rebholz-Schuhmann et al. (2010). First, instead of using verbs as the names of interaction words of PPI, we utilise interaction relation words that contain verbs and nouns. Second, the intrinsic differences between the two methods lie in how the relation words are derived from. Compared with the method in Rebholz-Schuhmann et al. (2010), we utilise a small amount of manually selected verbs as seeding words and extend these words by using WordNet automatically and the whole expanded process is relatively corpus-independent. Our method can cover the interaction relation words widely, including its conjugation, while their selected verbs are limited to some specific corpora. If the relation words are unseen in the specific corpus, their method could not find the PPI expressed by these words, while our method can. Finally, as for the importance of the interaction relation words, although their method gave high prediction capacity of verbs, the relations of verbs are isolated and flat. However, our method defines a hierarchical structure and gives a normalisation step to handle the relations of these words, which is important for the scalability of our PPI extraction.

RelEx (Fundel et al., 2007) extracts relations based on NLP, which produces dependency parse trees and it applies three simple rules to these trees. Our method is different. First, there are no explicit descriptions on how the relation words are obtained by RelEx, while we construct the IRO to represent the relation words. Second, RelEx is a bottom-up approach, which locates the protein name initially and then uses rules to extract PPIs. In contrast, our method is a top-down approach, identifying the interaction relation words initially and then constructing the interaction path of the proteins based on a dependency tree. Third, in our method, the six types of dependency trees reflect the location distributions of proteins relative to the interaction relation words rather than the fixed extraction rules. Two of the three rules in RelEx, '*effector-relation-effectee*' and '*relation-of- effectee- by-effector'*, have the same type of the rules in Figure 6. Finally, our method can capture the interaction relations between multiple proteins rather than just two proteins by RelEx. For example, in the sentence form '*A and B interact with C and D*', our method can extract four protein interaction pairs, {(A, C), (A, D), (B, C), (B, D)}, rather than only one pair {(C, D)}.

4.2 Complexity analysis

The computational complexity of our method is O ($N*logM$), where M is the size of IRO and N is the number of words in the sentence. If M is a constant, the computational complexity equals to O (N) approximately.

As proven by Collins and Duffy (2001), the computational complexity of subset tree kernel is quadratic in the number of tree nodes. Moschitti (2006) reported that the partial tree kernel also has a quadratic complexity. In Tikk et al. (2010), the authors stated that the computational complexity of the Edit kernel and the APG kernel are quadratic and cubic, respectively. As shown in Zelenko et al. (2003), the computational complexity of sparse subtree kernels is O ($N*M^3$), where N and M ($N > M$) are the sizes of two partitions of a sentence. As described in the work of Hastie et al. (2001), the basic kernel function methods have an initial cost of at least O ($N*(logN)^2 + (logN)^3$). We can see that our method has a lower computational complexity than most of the state-of-the-art kernel-based methods.

5 Conclusions

In this paper, IRO is introduced and applied to relation sentence classification and PPI extraction.

IRO is used with a binary classifier for identifying whether sentences are relation ones or not. By increasing the weights of relation words in the classifier, the performance of the binary classification improves significantly. IRO is also applied to guide PPI extraction by building sentence dependency parse tree based on the Stanford typed dependency, which can well capture the syntactic relations between proteins. Our dependency tree filtering and PPI extraction method can not only cover the potential protein pairs by Cartesian products and Combinations comprehensively, but also filter out non-relation pairs effectively. Extensive evaluations and detailed analysis have shown the effectiveness of IRO on relation sentences classification and PPI extraction.

Since our method relies on the quality of a sentence parser, the PPI extraction performance can be further improved by exploring advanced parser technologies. As there are many negative relations in the sentence dependency parse trees, the polarity of relations will be studied in the future. We will also explore the protein interaction networks in the biomedical literature resources to improve our method of PPI extraction.

Acknowledgements

We thank the anonymous reviewers for their constructive comments on the paper. This research was supported in part by the National Natural Science Foundation of China (No. 60774096) and the National Key Technology R&D Program of China (No. 2008BAK49B05). This work was also supported in party by the National Science Foundation of China (No. 60803107).

References

Ahmed, S.T., Chidambaram, D., Davulcu, H. and Baral, C. (2005) 'IntEx: a syntactic role driven protein-protein interaction extractor for bio-medical text', *Proceedings of the ACL-ISMB Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics*, Detroit, Michigan, 1641492: Association for Computational Linguistics, pp.54–61.

Aranda, B., Achuthan, P., Alam-Faruque, Y., Armean, I., Bridge, A., Derow, C., Feuermann, M., Ghanbarian, A.T., Kerrien, S., Khadake, J., Kerssemakers, J., Leroy, C., Menden, M., Michaut, M., Montecchi-Palazzi, L., Neuhauser, S.N., Orchard, S., Perreau, V., Roechert, B., van Eijk, K. and Hermjakob, H. (2009) 'The IntAct molecular interaction database in 2010', *Nucleic Acids Research*, Vol. 38, (Database issue), pp.D525–D531.

Bader, G.D., Donaldson, I., Wolting, C., Ouellette, B.F., Pawson, T. and Hogue, C.W. (2001) 'BIND–the biomolecular interaction network database', *Nucleic Acids Research*, Vol. 29, No. 1, pp.242–245.

Bunescu, R., Ge, R., Kate, R., Marcotte, E., Mooney, R., Ramani, A. and Wong, Y. (2005) 'Comparative experiments on learning information extractors for proteins and their interactions', *Artificial Intelligence in Medicine*, Vol. 33, No. 2, pp.139–155.

Bunescu, R. and Mooney, R. (2006) 'Subsequence Kernels for Relation Extraction', in Weiss, Y., Schölkopf, B. and Platt, J. (Eds.): *Advances in Neural Information Processing Systems 18*, MIT Press, pp.171–178.

Chen, X-W., Han, B., Fang, J. and Haasl, R.J. (2008) 'Large scale protein-protein interaction prediction using novel kernel methods', *Int. J. Data Min. Bioinformatics*, Vol. 2, N. 2, pp.145–156.

Chowdhary, R., Zhang, J. and Liu, J. (2009) 'Bayesian inference of protein-protein interactions from biological literature', *Bioinformatics*, Vol. 25, No. 12, pp.1536–1542.

Cohen, A. and Hersh, W. (2006) 'The TREC 2004 genomics track categorization task: classifying full text biomedical documents', *Journal of Biomedical Discovery and Collaboration*, Vol. 1, p.4.

Collins, M. and Duffy, N. (2001) *Convolution Kernels for Natural Language*, translated by Dietterich, T., Becker, S. and Ghahramani, Z., MIT Press, pp.625–632.

de Marnee, M-C. and Manning, C. (2010) *Stanford Typed Dependencies Manual*, Available at: http://nlp.stanford.edu/software/dependencies_manual.pdf

De Marneffe, M., MacCartney, B. and Manning, C. (2006) 'Generating typed dependency parses from phrase structure parses.' *Proceedings of LREC*-06 2006, Vol. 319, pp.449–454.

De Las Rivas, J. and Fontanillo, C. (2010) 'Protein–protein interactions essentials: key concepts to building and analyzing interactome networks', *PLoS Comput Biol*, Vol. 6, No. 6, p.e1000807.

Fan, R-E., Chang, K-W., Hsieh, C-J., Wang, X-R. and Lin, C-J. (2008) 'LIBLINEAR: A library for large linear classification', *Journal of Machine Learning Research*, Vol. 9, pp.1871–1874.

Fayruzov, T., Cock, M.D., Cornelis, C. and Hoste, V. (2008) 'The role of syntactic features in protein interaction extraction', *Proceeding of the 2nd International Workshop on Data and Text Mining in Bioinformatics*, Napa Valley, California, USA, 1458463: ACM, pp.61–68.

Fundel, K., Kuffner, R. and Zimmer, R. (2007) 'RelEx-Relation extraction using dependency parse trees', *Bioinformatics*, Vol. 23, No. 3, pp.365–371.

Gusfield, D. (2007) *Algorithms on Strings, Trees and Sequences : Computer Science and Computational Biology*, Cambridge Univ. Press.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. and Witten, I. (2009) 'The WEKA data mining software: an update', *SIGKDD Explorations*, Vol. 11, No. 1, pp.10–18.

Hastie, T., Tibshirani, R. and Friedman, J. H. (2001) *The elements of statistical learning: data mining, inference and prediction*, Springer.

Hoffmann, R. and Valencia, A. (2005) 'Implementing the iHOP concept for navigation of biomedical literature', *Bioinformatics (Oxford, England)*, Vol. 21, Suppl 2.

Howe, D.C. (2010) RiWordnet http://www.rednoise.org/rita/, [online], available: [Accessed: 06 Nov 2010].

Kim, J., Ohta, T., Pyysalo, S., Kano, Y. and Tsujii, J.i. (2009) *Overview of BioNLP'09 Shared Task on Event Extraction*, translated by Association for Computational Linguistics, pp.1–9.

Kim, S., Shin, S-Y., Lee, I-H., Kim, S-J., Sriram, R. and Zhang, B-T. (2008) 'PIE: an online prediction system for protein-protein interactions from text', *Nucl. Acids Res*., Vol. 36(suppl_2), pp.W411–415.

Krallinger, M., Leitner, F., Penagos, C. and Valencia, A. (2008) 'Overview of the protein-protein interaction annotation extraction task of BioCreative II', *Genome Biology*, Vol. 9(Suppl 2), p.S4.

Krallinger, M., Rodriguez-Penagos, C., Tendulkar, A. and Valencia, A. (2009) 'PLAN2L: a web tool for integrated text mining and literature-based bioentity relation extraction', *Nucleic Acids Research*, Vol. 37 (Web Server issue):W160-5.

Leitner, F., Mardis, S., Krallinger, M., Cesareni, G., Hirschman, L. and Valencia, A. (2010) 'An overview of BioCreative II.5', *IEEE/ACM Transactions on Computational Biology and Bioinformatics / IEEE, ACM*, Vol. 7, No. 3, pp.385–399.

Liu, B., Qian, L., Wang, H. and Zhou, G. (2010) 'Dependency-driven feature-based learning for extracting protein-protein interactions from biomedical text', *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, Beijing, China, 1944653: Association for Computational Linguistics, pp.757–765.

Miller, G. (1995) 'WordNet: a lexical database for English', *Commun. ACM*, Vol. 38, No. 11, pp.39–41.

Miwa, M., Sætre, R., Miyao, Y. and Tsujii, J.i. (2009) 'Protein-protein interaction extraction by leveraging multiple kernels and parsers', *International Journal of Medical Informatics*, Vol. 78, No. 12, pp.e39–e46.

Moschitti, A. (2006) 'Efficient convolution kernels for dependency and constituent syntactic trees', in Fürnkranz, J., Scheffer, T. and Spiliopoulou, M. (Eds.): *Machine Learning: ECML 2006*, Springer Berlin / Heidelberg, pp.318–329.

Nikitin, A., Egorov, S., Daraselia, N. and Mazo, I. (2003) 'Pathway studio--the analysis and navigation of molecular networks', *Bioinformatics*, Vol. 19, No. 16, pp.2155–2157.

Novichkova, S., Egorov, S. and Daraselia, N. (2003) 'MedScan, a natural language processing engine for MEDLINE abstracts', *Bioinformatics*, Vol. 19, No. 13, pp.1699–1706.

Polajnar, T., Rogers, S. and Girolami, M. (2011) 'Protein interaction detection in sentences via gaussian processes: A preliminary evaluation', *Int. J. Data Min. Bioinformatics*, Vol. 5, No. 1, pp.52–72.

Pollard, C. and Sag, I. (1994) *Head-Driven Phrase Structure Grammar*, University of Chicago press and CSLI publications.

Pyysalo, S., Ginter, F., Heimonen, J., Bjorne, J., Boberg, J., Jarvinen, J. and Salakoski, T. (2007) 'BioInfer: a corpus for information extraction in the biomedical domain', *BMC Bioinformatics*, Vol. 8, No. 1, p.50.

Rebholz-Schuhmann, D., Jimeno-Yepes, A., Arregui, M. and Kirsch, H. (2010) 'Measuring prediction capacity of individual verbs for the identification of protein interactions', *J. of Biomedical Informatics*, Vol. 43, No. 2, pp.200–207.

Rudniy, A., Song, M. and Geller, J. (2010) 'Detecting duplicate biological entities using shortest path edit distance', *Int. J. Data Min. Bioinformatics*, Vol. 4, No. 4, pp.395–410.

Saetre, R., Sagae, K. and Tsujii, J.I. (2008) 'Syntactic features for protein-protein interaction extraction', *Proceedings of LBM'07*, Vol. 319.

Sun, C., Lin, L., Wang, X. and Guan, Y. (2007) 'Using Maximum Entropy Model to Extract Protein-Protein Interaction Information from Biomedical Literature'in, pp.730–737.

Taylor, R.C., Singhal, M., Daly, D.S., Gilmore, J., Cannon, W.R., Domico, K., White, A. M., Auberry, D.L., Auberry, K.J., Hooker, B.S., Hurst, G., McDermott, J.E., McDonald, W.H., Pelletier, D.A., Schmoyer, D. and Wiley, H.S. (2009) 'An analysis pipeline for the inference of protein-protein interaction networks', *Int. J. Data Min. Bioinformatics*, Vol. 3, No. 4, pp.409–430.

Tikk, D., Thomas, P., Palaga, P., Hakenberg, J. and Leser, U. (2010) 'A comprehensive benchmark of kernel methods to extract protein–protein interactions from literature', *PLoS Comput Biol*, Vol. 6, No. 7: e1000837. doi:10.1371/journal.pcbi.1000837

Van Landeghem, S., Abeel, T., Saeys, Y. and Van de Peer, Y. (2010) 'Discriminative and informative features for biomolecular text mining with ensemble feature selection', *Bioinformatics (Oxford, England)*, Vol. 26, No. 18, pp.i554–i560.

Xenarios, I., Salwínski, L., Xiaoqun, J., Higney, P., Kim, S-M. and Eisenberg, D. (2002) 'DIP, the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions', *Nucleic acids research*, Vol. 30, No. 1, pp.303–305.

Yang, Z., Lin, H. and Wu, B. (2009) 'BioPPIExtractor: A protein-protein interaction extraction system for biomedical literature', *Expert Syst. Appl.*, Vol. 36, No. 2, pp.2228–2233.

Zelenko, D., Aone, C. and Richardella, A. (2003) 'Kernel methods for relation extraction', *J. Mach. Learn. Res.*, Vol. 3, pp.1083–1106.