View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Elsevier - Publisher Connector

Journal of Biomedical Informatics 43 (2010) 88–96

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

# BioPPISVMExtractor: A protein–protein interaction extractor for biomedical literature using SVM and rich feature sets

Zhihao Yang *, Hongfei Lin, Yanpeng Li

*Department of Computer Science and Engineering, Dalian University of Technology, Dalian 116023, China*

ARTICLE INFO

*Article history:*
Received 17 December 2008
Available online 23 August 2009

*Keywords:*
Text mining
Information extraction
Protein–protein interaction
Conditional random fields
Support vector machines

ABSTRACT

Protein–protein interactions play a key role in various aspects of the structural and functional organization of the cell. Knowledge about them unveils the molecular mechanisms of biological processes. However, the amount of biomedical literature regarding protein interactions is increasing rapidly and it is difficult for interaction database curators to detect and curate protein interaction information manually. This paper presents a SVM-based system, named BioPPISVMExtractor, to identify protein–protein interactions in biomedical literature. This system uses rich feature sets including word features, keyword feature, protein names distance feature and Link path feature for SVM classification. In addition, the Link Grammar extraction result feature is introduced to improve the precision rate. Experimental evaluations with other state-of-the-art PPI extraction systems tested on the DIP corpus indicate that BioPPISVMExtractor can substantially improve recall at the cost of a moderate decline in precision.

© 2009 Elsevier Inc. All rights reserved.

## 1. Introduction

Protein–protein interactions (PPI) play a key role in various aspects of the structural and functional organization of the cell. Knowledge about them unveils the molecular mechanisms of biological processes. A number of databases such as MINT [1], BIND [2], SwissProt [3], and DIP [4] have been created to store protein interaction information in structured and standard formats. However, the amount of biomedical literature regarding protein interactions is increasing rapidly and it is difficult for interaction database curators to detect and curate protein interaction information manually. Thus, most of the protein interaction information remains hidden in the unstructured text of the published papers. Therefore, automatic extraction of protein interaction information from biomedical literature has become an important research area.

Existing PPI extraction works can be roughly divided into three categories: manual pattern engineering approaches, grammar engineering approaches and machine learning approaches.

Manual pattern engineering approaches define a set of rules for possible textual relationships, called patterns, which encode similar structures in expressing relationships. The SUISEKI system of Blaschke uses regular expressions, with probabilities that reflect the experimental accuracy of each pattern to extract interactions into predefined frame structures [5]. Ono et al. manually defined a set of rules based on syntactic features to preprocess complex sentences, with negation structures considered as well [6]. Their method achieved good performance with a recall rate of 85% and a precision rate of 84% for *Saccharomyces cerevisiae* (yeast) and *Escherichia coli*. Leroy and Chen employed preposition-based parsing to generate templates, which achieved a template precision of 70% when processing literature abstracts [7]. The BioRAT system uses manually engineered templates that combine lexical and semantic information to identify protein interactions [8]. Such manual pattern engineering approaches for protein interaction extraction require labor-intensive and skill-dependent pattern engineering.

Grammar engineering approaches use manually generated specialized grammar rules that perform a deep parse of the sentences. These approaches can be further divided into two types, based on the complexity of the linguistics methods, as shallow (or partial) parsing or deep (or full) parsing. Shallow parsing techniques aim to recover syntactic information efficiently and reliably from unrestricted text, by sacrificing completeness and depth of analysis. Sekimizu et al. used shallow parser, EngCG, to generate three kinds of tags, such as syntactic, morphological, and boundary tags [9]. Based on the tagging results, subjects and objects were recognized for the most frequently used verbs in a collection of abstracts which were believed to express the interactions between proteins, genes. Pustejovsky et al. targeted "inhibit" relations in the text and built a finite-state automata (FSA) to recognize these relations [10]. Leroy et al. used a shallow parser to automatically capture the relationships between noun phrases in free text [11].

Deep parsing techniques analyze the entire sentence structure, which normally achieve better performance but with increased

* Corresponding author. Address: Department of Computer Science and Engineering, Dalian University of Technology, No. 2 LingGong Road, ShaHeKou District, Dalian 116023, China. Fax: +86 0411 84706550.
*E-mail address:* yangzh@dlut.edu.cn (Z. Yang).

1532-0464/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jbi.2009.08.013

computational complexity. Temkin used a context free grammar that is designed specifically for parsing biological text [12]. In [13], a broad-coverage probabilistic dependency parser was used to identify sentence level syntactic relations between the heads of the chunks. The parser used a hand-written grammar combined with a statistical language model that calculates lexicalized attachment probabilities. Fundel et al. proposed RelEx based on the dependency parse trees to extract relations in biomedical texts [14]. It was applied on one million MEDLINE abstracts to extract gene and protein relations. About 150,000 relations were extracted with an estimated performance of both 80% precision and 80% recall. Recently, extraction systems have also used Link Grammar to identify interactions between proteins [15–16]. Their approach relies on various linkage paths between named entities such as gene and protein names. For example, the IntEx system extracts interactions by analyzing the matching contents of syntactic roles and their linguistically significant combinations via Link Grammar [16].

Machine learning approaches for extracting protein interaction information have gained interest in the recent years. Marcotte's supervised learning text classification can decide PPI information which is mentioned in the text [17]. Donaldson et al. constructed PreBIND and Textomy – an information extraction system that uses support vector machines to evaluate the importance of protein–protein interactions [18]. Xiao et al. used Maximum Entropy models to combine diverse lexical, syntactic and semantic features for PPI extraction [19]. Zhou et al. employed a semantic parser using the Hidden Vector State (HVS) model for protein–protein interactions which can be trained using only lightly annotated data whilst simultaneously retaining sufficient ability to capture the hierarchical structure [20]. Airola et al. proposed a graph kernel based approach which captures the information in unrestricted dependency graphs for the automated extraction of protein–protein interactions [21].

A wide range of results have been reported for the PPI extraction systems, but differences in evaluation resources, metrics and strategies make direct comparison of the numbers presented problematic [21]. Further, the results gained from the BioCreative II evaluation, where the best performing system achieved a 29% *F*-score [22], suggest that the problem of extracting binary protein–protein interactions is far from solved.

Currently, several publicly available PPI corpora such as AImed [23], BioInfer [24] and HPRD50 [25], IEPA [26] and LLL [27], have been developed to train and evaluate PPI extraction methods. Blaschke and Valencia recommend using DIP as a way of evaluating biological IE systems, because it represents a realistic problem of practical interest to biological researchers [28]. IE researchers can use their systems to extract protein–protein interactions, and then compare these with the records in DIP. The BioRAT system uses templates and achieves a recall of 20.31% and a precision of 55.07% on 389 interactions from the DIP database corresponding to 229 MEDLINE abstracts [8]. On the same dataset, the IntEx system using Link Grammar to identify interactions between proteins achieves a recall of 26.94% and a precision of 65.66% [16]. However, both BioRAT and IntEx adopt the dictionary-based protein name recognition method, which leads to poor recall performance (20.31% and 26.94% respectively) since the dictionary-based method depends badly on the size and quality of the protein name dictionary. For example, the recall errors generated in protein name recognition account for about 60% and 45% respectively in BioRAT and IntEx. The BioPPIExtractor system, like the IntEx system, also uses a Link Grammar parser to identify the syntactic roles in sentences and then extracts interactions from these syntactic roles [29]. By introducing a CRF-based protein name recognition method, BioPPIExtractor achieves a much better recall of 41.84% and a still good precision of 55.41%.

However, the recall performance of existing systems tested on the DIP dataset (including manual pattern engineering approach (BioRAT) and grammar engineering approaches (IntEx and BioPPIExtractor)) are still not satisfactory (BioRAT (20.31%), IntEx (26.94%) and BioPPIExtractor (39.80%)). If the work of database curators will be supported by PPI automatic extraction systems, we speculate that, recall is a more concerned performance metric than precision in that high recall means as much PPI information as possible can be automatically extracted for further identification and a recall of less than 40% is far from being good enough.

Our work aims to further boost the recall performance on DIP dataset while keeping an acceptable precision. In this paper we present a SVM-based fully automated PPI extraction system – BioPPISVMExtractor. The system first applies a CRFs-based method to improve the performance of protein name recognition [30] (discussed later in Section 2.2). Then it uses rich feature sets including word features, keyword feature, protein names distance feature and Link path feature for SVM classification, which substantially improves the recall. In addition, the Link Grammar extraction result feature is introduced to improve the precision. Our experimental results show that BioPPISVMExtractor system can achieve a much better recall than other systems on the same dataset while having a still good precision.

The remaining part of this paper is organized as follows: Section 2 describes the architecture of BioPPISVMExtractor system and its processing stages. Section 3 presents and discusses the experimental results. Section 4 offers some concluding remarks.

## 2. Methods

The system architecture of BioPPISVMExtractor is shown in Fig. 1. In this system the IEPA corpus [26] is used as the training set for SVM classifier and the DIP corpus is used as the test set. The trained SVM classifier is used to identify which protein pairs in a sentence have a biologically relevant relationship between them. In other words, we model the extraction as a binary classification problem. BioPPISVMExtractor consists of the following processing stages to extract interaction information: pronoun resolution, protein name recognition, features extraction and SVM classification. The details are described in the following sections.

### 2.1. Pronoun resolution

Extracting PPI from text should take into account the resolution of pronominal references to proteins since interactions are often specified through these references. Our anaphora resolution module currently focuses on third person pronouns and reflexives since the first and second person pronouns are frequently used to refer to the authors of the papers. In our pronoun resolution module, noun and noun phrase in text are identified using GENIA Tagger which is specifically tuned for biomedical text such as MEDLINE abstracts and achieves an *F*-score of 98.20% on GENIA corpus [31,32]. Then the nearest noun (phrase) that matches the number of the pronoun (singular or plural) is considered as the referred phrase. To determine a noun (phrase)'s number, we define a set of rules based on morphological knowledge. In most cases, nouns ending in *-s* are plural nouns. Some nouns have no obvious plural form, for instance, *sheep*, *deer*, *fish*, *cattle*, *vermin* and so on. There are also other nouns which have irregular plurals, although they are few in number: *mice*, *teeth*, *geese* and *children*. There are a number of nouns that look plural but are treated as singulars: *news*, *Physics* and so on. In other cases, the nouns are singular nouns.

### 2.2. Protein name recognition

In biomedical domain protein name recognition remains a challenging task due to the irregularities and ambiguities in protein

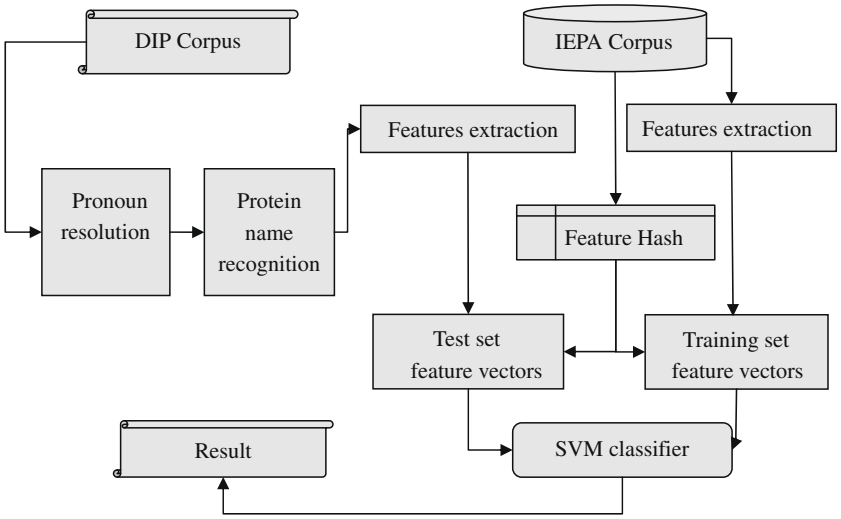


**Fig. 1.** System architecture of BioPPISVMExtractor.

nomenclature. The performance of the state of art is about 80% in *F*-score (using exact matching) which is far below the one of NER in the general domain. In BioPPISVMExtractor, a CRFs-based protein name recognition method is applied which can achieve fairly good performance [30].

Protein name recognition can be thought of as a sequence segmentation problem: each word is a token in a sequence to be assigned a label (e.g., protein or other). Conditional random fields are undirected statistical graphical models, a special case of which is a linear chain that corresponds to a conditionally trained finite-state machine. Such models are well suited to sequence analysis. They have recently been applied to the more limited task of finding gene and protein mentions with promising early results [33].

Feature based statistical models like CRFs reduce the problem to finding an appropriate feature set. The following features are used in our CRFs model:

(1) *Surface word features:* We use words themselves as features. All the words are lower-cased to improve the recall, for example, "JAK2" and "Jak2" are all gene names and the loss of information can be compensated through its combination with other features.
(2) *Orthographic features:* Orthographic information is indicative to the class of biomedical named entity. We use regular expressions to extract several orthographic patterns (e.g., capture capitalization and digitalization) from each token of the text and assign the token a binary feature. These features are especially useful to recognize unknown terms.
(3) *Prefix/suffix features:* Many biomedical entities have certain prefix or suffix, such as "antiglobulin" or "insulin". For each token the three and four characters' prefix and suffix are used as features. For example, for the token "antiglobulin", the feature tags will be "prefix3 = ant", "prefix4 = anti", "suffix3 = lin" and "suffix4 = ulin". These features are all binary types.
(4) *Word shape features:* Word shapes refer to mappings of each word to a simplified representation that encodes attributes such as its length and whether it contains capitalization, numerals, Greek letters, and so on. For example, capital letters are replaced with 'A', lowercase letters with 'a', digits with '0', and all other characters with 'x'. Thus "Varicella-zoster" would become Xx-xxx, and "CPA1"would become XXX0.
(5) *Compound features:* To model local context simply, neighboring words in the window [−1, 1] are also added as features.
(6) *Part-of-speech* (POS) features: POS may provide useful evidence about the boundaries of biomedical entity names. Here, GENIA Tagger is used to output POS tags.
(7) *Keyword features:* Some words occur more frequently in the biomedical named entity names. These words (we called keywords) such as "factor", "receptor", "site", etc. can help to identify entity names. We automatically extract unigram and bigram keywords which occur more than 20 times from the training data.
(8) *Boundary word features:* Most frequent boundary terms (including 1-gram and 2-gram) that appear in training data more than 5 times are listed. If the text matches terms in the list, it will be assigned this feature. It may be overlapped with word feature, but in our experiment, we found that it could slightly improve the performance.

Trained on BioCreative 2004 task 1A training set, our CRFs model achieves an *F*-score of 81.32% (using relax matching) on BioCreative test set. The performance is improved to 83.7% via the exploitation of some contextual cues including bracket pair, heuristic syntax structure and interaction words cue.

### 2.3. SVM model

The heart of the BioPPISVMExtractor system is a SVM classifier that is trained to recognize protein–protein interactions in biomedical texts. The foundations of support vector machines (SVM) have been developed by Vapnik and are gaining popularity due to many attractive features, and promising empirical performance [34]. The PreBIND system uses SVM to identify the existence of protein interactions in abstracts [18]. Words and word bigrams are used as binary features. This system is also tested with the Naive Bayes classifier, but SVM is reported to perform better. Sugiyama et al. extracted features from the sentences based on the verbs and nouns in the sentences [35]. They constructed k-nearest neighbor, decision tree, neural network, and SVM classifiers by using these features. They also reported that the SVM classifier performs the best.

SVMs are binary classifiers for a set of training data $(x_i, y_i)$, $i = 1, \cdots, n, x_i \in R^N, y_i \in \{+1, -1\}$ where $x_j$ is a feature vector of the $j$th training sample, and $y_j$ is the class label associated with the $j$th training sample. The decision function is defined by

$$y(x) = \mathrm{sgn}\left(\sum_{j \in SV} y_j \alpha_j \phi(\mathrm{x}_j) \cdot \phi(\mathrm{x}) + b\right) \quad (1)$$

where $\phi$ is a nonlinear mapping function from $R^N$ to $R^H$ ($N<<H$), $\alpha_j, b \in R$ , $\alpha_j \geqslant 0$, and SV is a set of support examples. The mapping function $\phi$ should be designed such that all training examples are linearly separable in $R^H$ space. SVMs take a maximal margin strategy in that the parameters are chosen so that the minimum distance between examples and the separating hyperplane (i.e., margin) is maximized.

The kernel we applied in SVM model is a linear kernel. One of the advantages of the SVM with a linear kernel is that it can handle high dimensional data effectively since it compares the "active" features rather than the complete dimensions [36,37]. We can therefore impose richer feature types upon each training example to enhance system performance. According to [38], the richer feature set had shown to be more effective than the simple feature set. Another advantage of linear kernel SVM is its low training and testing time costs. Keerthi and DeCoste had reported that the training time of linear kernel SVM can be reduced to linear time [39]. In addition, using linear kernel SVM only penalty parameter C needs to be adjusted in the algorithm, which is usually set as a constant in applications.

In our experiments, the SVM-light package [40] was used. The penalty parameter C in setting the SVM is a very important parameter since it controls the tradeoff between the training error and the margin. The SVM-light package did an excellent job at setting the default value for this parameter. In our experiments the parameter was left as default value since we observed that other manually determined values of this parameter in fact led to worse performance of the SVM when compared with the default one.

### 2.4. Feature selection

In our experiments the following features are exploited for SVM classification:

(1) Words

The following three sets of word features are used.

- Words from two protein names: these features include all words that appear in two protein names.
- Words between two protein names: these features include all words that are located between two protein names.
- Words surrounding two protein names: These features include left $N$ words of the first protein name and right $N$ words of the second protein name. $N$ is the number of surrounding words considered which is set to be three in our experiments. All words are treated as bag-of-word. That is, the order of these words is not considered.

(2) Protein name distance

The shorter the distance (the number of words) between two protein names is, the more likely the two proteins have interaction relation. Therefore the distance between two protein names is chosen as a feature. If there are less than three words between two proteins, the feature value is set to "DISLessThanThree"; if there are more than three words but less than six words between two proteins, the feature value is set to "DISBetweenThreeSix". The other feature values include "DISBetweenSixNine", "DISBetweenNineTwelve" and "DISMoreThanTwelve".

(3) Keyword

The existence of an interaction keyword (the verb expressing protein interaction relation such as "bind", "interact", "inhibit", etc.) between two protein names or among the surrounding words of two protein names often implies the existence of the protein–protein interaction. Therefore, the keyword is chosen as a feature. To identify the keywords in texts, we built an interaction keyword list of about 500 entries manually, which include the interaction verbs and their variants (for example, interaction verb "bind" has variants "binding" and "bound").

(4) Link path

Link Grammar was first introduced by Sleator and Temperley to simplify English grammar with a context free grammar [41]. The basic idea of Link Grammar is to connect pairs of words in a sentence with various links. Each word is viewed as a block with connectors coming out. There are various types of connectors, and connectors may point to the right or to the left. A link consists of a left-pointing connector connected with a right-pointing connector of the same type on another word. A valid sentence is one in which all the words are connected in some way.

If there is a Link path between two protein names, the two proteins tend to have interaction relation. Therefore the Link path is used as a binary feature. If there is a Link path between two protein names, the Link path feature value of the two proteins will is set to "Link_YES", otherwise, "Link_NO". The Link Grammar parser used in BioPPISVMExtractor was developed by Grinberg et al. [42]. The parser's dictionary can also be easily enhanced to produce better parses for biomedical text. Owing to the dictionary, the parser can recognize most words in biomedical domain.

(5) Link Grammar extraction result

Machine learning approaches usually can achieve better recall rates than other approaches since every pair of protein names in a sentence will be checked whether it has interaction relation. However, for the same reason, their precision rates will decrease since many false positives will be introduced. To alleviate the problem, the Link Grammar extraction result feature is introduced into our SVM feature set.

We have presented a grammar engineering-based PPI extraction system-BioPPIExtractor [29]. The BioPPIExtractor system achieves a fairly good precision of 55.41% by via of Link Grammar parsing. Therefore, the introduction of the Link Grammar extraction result information may improve the precision of the SVM-based PPI extraction approach. If a pair of protein names is extracted with Link Grammar parsing method, the Link Grammar extraction feature value of the pair will is set to "LinkExtracted_YES", otherwise, "LinkExtracted _NO".

For the sentence "we show here that recombinant *bovine prion protein* strongly interacts with the catalytic alpha/alpha' subunits of *protein kinase*," the extracted features are show in Table 1.

## 3. Experiment and discussion

### 3.1. Corpus

The training set used in our experiments is the Interaction Extraction Performance Assessment (IEPA) corpus provided by Iowa State University [26]. It consists of 303 abstracts retrieved from MEDLINE using ten queries (each query was an AND expression of two biochemical nouns) through PUBMED interface. Among these abstracts there are 336 positive instances (the protein pairs

**Table 1**
Features extracted from example sentence A.

| Feature names | Feature values |
|---|---|
| First protein name | p1_bovine, p1_prion, p1_protein |
| Second protein name | p2_protein, p2_kinase |
| Words between two protein names | b_strongly, b_interacts, b_with, b_the |
| Left words | l_here, l_that, l_recombine |
| Right words | r_ |
| Protein name distance | DISBetweenSixNine |
| Keyword | k_interacts |
| Link path | LinkYES |
| Link Grammar extraction result | LinkExtractedYES |

having interaction relation) and 308 negative instances (the protein pairs without interaction relation). All protein names have been tagged correctly in the IEPA corpus.

The test set in our experiments is the same dataset that was used for the BioPPIExtractor, BioRAT and IntEx evaluation so that the results are comparable. For evaluation, 394 interactions were identified from the DIP database such that both proteins participating in the interaction had SwissProt entries. These interactions correspond to 229 abstracts from the PubMed. Since we did not achieve the responding full papers, the BioPPISVMExtractor system was tested only on the abstracts.

### 3.2. Evaluation and results

We evaluated the results in a similar way to BioPPIExtractor, BioRAT and IntEx. The 1814 candidate interactions extracted by BioPPISVMExtractor were manually examined by a domain expert for precision and recall. Our domain expert manually compared the predictions made by BioPPISVMExtractor to the source DIP records to measure the recall. For each record in DIP, our domain expert searched through the output of BioPPISVMExtractor corresponding to the same Medline abstract, and checked to see if the interaction mentioned in DIP has been identified. Precision is harder to measure than recall, because we need an estimate of the number of false positives. If a record produced by BioPPISVMExtractor is not found in DIP, it could be that (a) it is a false-positive example, reducing the precision of BioPPISVMExtractor; or (b) the record is missing from DIP. The latter case consists of interactions that are mentioned in Medline abstract, but have yet to be added to DIP. Our domain expert manually checked each interaction extracted by the system. If the protein pair in an interaction has a biologically relevant relationship, the interaction is used as a true positive, whether or not the information is in DIP. Results are given as *F*-scores (the harmonic mean of precision and recall, defined as $F = (2PR)/(P + R)$ where $P$ denotes precision and $R$ recall).

We also evaluated the performances using the *area under the receiver operating characteristics curve* (AUC) measure [43]. Formally, AUC can be defined as

$$\mathrm{AUC} = \frac{1}{n_p * n_n} \sum_{i=1}^{n_p} \sum_{j=1}^{n_n} C(s(P_i), s(N_j)) \tag{2}$$

where $n_p$ and $n_n$ are the number of positive and negative cases, $s$ is the score of a case, and $P$ and $N$ are the sets of positive and negative test cases. When using a set of test data to estimate the probability that a randomly selected positive case will receive a higher score than a randomly selected negative case, we compare the scores assigned by a model to each case in the test set. The function $C(s_p, s_n)$ for comparing $s_p$, the score of a positive case, with $s_n$, the score of a negative case, is defined as

$$C(s_p, s_n) = \begin{bmatrix} 1 & \text{if} & s_p > s_n \\ 0.5 & \text{if} & s_p = s_n \\ 0 & \text{if} & s_p < s_n \end{bmatrix} \tag{3}$$

AUC has the important property that it is invariant to the class distribution of the used dataset and has been advocated to be used for performance evaluation in the machine learning community [44]. According to [21], there is a critical weakness of the *F*-score metric in comparisons involving different corpora (for example, the fraction of true interactions out of all candidates is 50% in the LLL corpus but only 17% in AImed). By contrast to the large differences in performance measured using *F*-score, it is found that for the distribution-invariant AUC measure the performance for all of the corpora falls in the range of 80–85%. The results provide an argument in favor of applying the AUC metric instead of, or in addition to, *F*-score.

The classification performances of different features and their combinations are shown in Table 2 (the recall, precision and *F*-score values are achieved with the optimal threshold values obtained from the 10-fold cross-validations using the IEPA training set). Using only word feature a recall of 81.54% and a precision of 27.68% are achieved. With the introduction of protein names distance feature, keyword feature, Link path feature and Link Grammar extraction result feature, the *F*-score and AUC performances are steadily improved. Among others, the keyword feature contributes most to the performance improvement (8.35 percentage points' increase in *F*-score and 11 percentage points' increase in AUC).

Table 3 presents the evaluation results as compared with BioPPIExtractor, BioRAT and IntEx. In Table 3 the recalls, different from those in Table 2 (which are used to evaluate the classification performance of different features), are calculated as the ratio of extracted DIP interactions to 394 (the sum of all interactions selected from the DIP database). The recall of BioPPISVMExtractor (71.83%) is much higher than BioPPIExtractor (41.62%), BioRAT (20.31%) and IntEx (26.94%). The reason is as follows: compared with BioPPIExtractor, BioPPISVMExtractor checks whether every pair of proteins in a sentence has interaction relation and, therefore, extracts more interactions; compared with BioRAT and IntEx, besides the reason just-mentioned, BioPPISVMExtractor (like BioPPIExtractor) applies a CRFs-based protein name recognition method with much better performance than those of BioRAT and IntEx and extracts more interactions. The ability of achieving higher recall is the advantage of BioPPISVMExtractor since as much PPI information as possible can be automatically extracted for the interaction database curators' further identification. Compared with others, the precision of BioPPISVMExtractor is the lowest (49.28%). The reason, as discussed in Section 2.4, is that many false positives are introduced by SVM-based method. However, 49.28% is still an acceptable precision.

In the term of *F*-score, the performance of BioPPISVMExtractor (58.46%) is much better than those of BioPPIExtractor (47.53%), BioRAT (29.68%) and IntEx (38.20%) because of its much higher recall.

In practice, it has been found that recall and precision are inversely related to each other [45], so that an increase in the recall generally is accompanied by a decrease in precision, and vice versa. If we value precision, we should set the threshold high; if we value recall, we should set the threshold low. The precision–recall curve in Fig. 2 shows the relationship between precision and recall of the BioPPISVMExtractor system. The figure shows data as the threshold of distance from the test example to the hyperplane is varied. In our experiments, the manually selected threshold scores range from the lowest score of test examples to a high value (less than the highest score) with an interval of 0.1. In general, if the threshold is set higher, a higher precision and a lower recall will be

**Table 2**
Classification performances of different features and their combinations in BioPPISVMExtractor tested on DIP corpus (measured with precision, recall, *F*-score and AUC). The * indicates the corresponding feature is used.

| Feature type | | | | | |
|---|---|---|---|---|---|
| Words | * | * | * | * | * |
| Protein names distance | | * | * | * | * |
| Keyword | | | * | * | * |
| Link path | | | | * | * |
| Link Grammar extraction result | | | | | * |
| Recall (%) | 81.54 | 80.77 | 74.61 | 70.23 | 70.04 |
| Precision (%) | 27.68 | 28.93 | 43.49 | 47.12 | 49.28 |
| *F*-score (%) | 41.32 | 42.60 | 54.95 | 56.40 | 57.85 |
| AUC (%) | 66.9 | 67.2 | 78.2 | 80.4 | 82.1 |

**Table 3**
Performance comparison with BioPPIExtractor, IntEx and BioRAT tested on DIP corpus (measured with precision, recall and *F*-score). The results of BioPPIExtractor, IntEx and BioRAT are from [29], [16] and [8] respectively. The bold values are the important evaluation metrics including recall, precision and F-score.

| | BioPPISVMExtractor | | BioPPIExtractor | | IntEx | | BioRAT | |
|---|---|---|---|---|---|---|---|---|
| | Cases | Percent | Cases | Percent | Cases | Percent | Cases | Percent |
| Recall | 283 | **71.83** | 164 | **41.62** | 142 | **26.94** | 79 | **20.31** |
| No Recall | 111 | 28.17 | 230 | 58.38 | 385 | 73.06 | 310 | 79.69 |
| Totals | 394 | 100.00 | 394 | 100.00 | 527 | 100.00 | 389 | 100.00 |
| Correct | 894 | **49.28** | 543 | **55.41** | 262 | **65.66** | 239 | **55.07** |
| Incorrect | 920 | 50.72 | 437 | 44.59 | 137 | 34.34 | 195 | 44.93 |
| Totals | 1814 | 100.00 | 980 | 100.00 | 399 | 100.00 | 434 | 100.00 |
| *F*-score | | **58.46** | | **47.53** | | **38.20** | | **29.68** |

The total interaction number (the fifth row) we obtained from Dr. David Corney (the author of [8]) is 394, a bit different from 389 (the number used in BioRAT evaluation). However, we do not know the reason why the number used in IntEx evaluation is 527 since we did not get into touch with the authors.

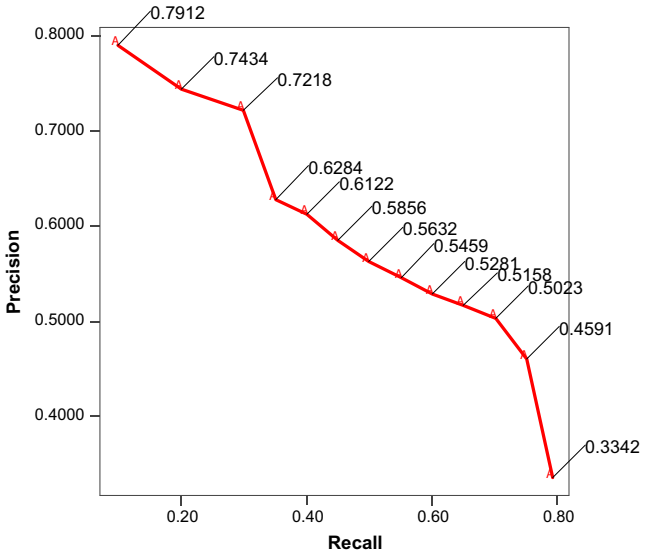


**Fig. 2.** Relationship between precision and recall.

achieved, and, if the threshold is set lower, a lower precision and a higher recall will be achieved. When recall is 10%, a high precision of 79.12% is achieved. Precision declines as recall increases. When recall reaches 75%, precision begins to drop sharply which means few positives will be returned with a recall more than 75%. When recall is 79.15% (the highest recall that BioPPISVMExtractor can achieve, more details are discussed in Section 3.4), the precision reaches its minimum value (33.42%), which is still almost 10 percentage points higher than the one of the co-occurrence (or *all-true*, co-occurrence of two biochemical names as an indicator of an interaction between them) extraction (23.52%). This shows that, with BioPPISVMExtractor system, interaction database curators can have a high recall with an acceptable precision.

### 3.3. Effectiveness of different features

#### 3.3.1. Word features

In the BioPPISVMExtractor system the words including words from, between and surrounding two protein names are used as the basic feature set since they include most contextual information about protein–protein interaction. Using only word features a recall of 81.54% and a precision of 27.68% are achieved. The *F*-score and AUC are 41.32% and 66.9% respectively.

#### 3.3.2. Protein names distance feature

The introduction of protein names distance feature is based on the idea that the shorter the distance between two protein names is, the more likely the two proteins have interaction relation. To the best of our knowledge, we are the first to use this feature in machine learning-based PPI extraction. Our experimental results show the introduction of protein names distance feature result in an increase in *F*-score by 1.7 percentage points and an increase in AUC by 0.3 percentage points.

#### 3.3.3. Keyword feature

The existence of interaction keywords in a sentence often implies the existence of the protein–protein interaction(s) in surrounding context. Our experimental results show that, among others, the keyword feature contributes most to the performance improvement (a decrease of 6.17 percentage points in recall and an increase of 14.56 percentage points in precision which results in an increase of 8.35 percentage points in *F*-score and an increase of 11 percentage points in AUC).

#### 3.3.4. Link path feature

If there is a Link path between two protein names, these two proteins tend to have an interaction relation. Ding et al. only used the existence or not of a Link path between a protein pair as a decision rule to extract interaction relation, achieving a recall of 87% and a precision of 61% on the IEPA corpus [15]. Like protein names distance feature, the Link path feature is first used in machine learning-based PPI extraction. In our experiment, the introduction of Link path feature results in an increase in *F*-score by 1.45 percentage points and an increase in AUC by 2.2 percentage points.

#### 3.3.5. Link Grammar extraction result feature

As described in Section 2.4, the Link Grammar parsing method extracts complete interactions by analyzing the matching contents of syntactic roles and can achieve a fairly good precision. The introduction of the Link Grammar extraction result feature may improve the precision of the SVM-based approach. Our experimental results show that the Link Grammar extraction result feature improves the precision with slight loss of the recall, resulting in better *F*-score and AUC (improved from 56.40% to 57.85% and 80.4% to 82.1% respectively).

The combined contributions of Link path feature and Link Grammar extraction result feature to performance improvement are an increase of 2.9 percentage points in *F*-score and an increase of 3.9 percentage points in AUC. Therefore, the combination of grammar engineering approaches and machine learning approaches may be a promising approach to achieve higher performance.

### 3.4. Error analysis

Confined to the complexity of natural language, extracting PPI interaction from biomedical literature remains a challenging task and it is difficult to achieve a satisfactory performance. A detailed

analysis of all types of recall errors of the BioPPISVMExtractor system is shown in Table 4.

DIP contains protein interactions from both abstracts and full text. Since the BioPPISVMExtractor system was tested only on the abstracts, the system missed out on some interactions that were only present in the full text. This accounts for about half of total recall errors (50.45%). If those interactions are excluded, BioPPISVMExtractor can have a recall of 83.63%.

In addition, recall errors occur in all the PPI extraction processing stages: pronoun resolution, protein name recognition and interaction extraction. Among others, the most errors are generated in interaction extraction stage (29.73%). The reason is that due to the complexity of the PPI interaction expression as well as the limited size of the training set, many PPI interactions are missed out. In addition, some errors happen in the procedure of extracting Link path feature and Link Grammar extraction result feature: Link Grammar parser itself may make some mistakes. For example, when dealing with too long sentences Link Grammar parser will get into "panic" model in which the parser can parse even very long sentences quickly, but with considerably reduced accuracy.

The recall errors generated in protein name recognition account for 15.32% of total recall errors. The numbers are about 60% and 45% in BioRAT and IntEx respectively since they adopt the dictionary-based protein name recognition method whose performance depends badly on the size and quality of the protein name dictionary. With the introduction of the CRFs-based protein name recognition method, the overall BioPPISVMExtractor PPI extraction performance is significantly improved. In addition, 4.5% recall errors are generated in pronoun resolution.

A detailed analysis of all types of precision errors is shown in Table 5. More than half precision errors (62.07%) are generated in interaction extraction stage. The reason is that confined to the complexity of the protein interaction expression as well as the quantity and quality of the training set, many false positives are generated. The other main cause of precision errors is protein name recognition. Many non-protein names are tagged leading to many false positives (accounting for 35.86%). Pronoun resolution also causes some precision errors (accounting for 2.07%).

### 3.5. Binary classification performance

The figures shown in Table 3 are the overall performances of BioPPISVMExtractor on DIP corpus including the ones of pronoun resolution, protein name recognition and binary classification (whether a pair of proteins in a sentence has interaction relation). To evaluate the binary classification performance of BioPPISVMExtractor separately, we tested it on IEPA, BioInfer, AImed, HPRD50 and LLL corpora in which all protein names have been tagged correctly and, therefore, pronoun resolution and protein name recognition are not needed. All the corpora were processed to a common format introduced in [46]. The unified format follows the standoff annotation principle, where the original sentence text is reserved and the entities are identified through character offsets. All results and discussion below concern these transformed versions of the corpora.

The performance comparison of our method with the graph kernel approach (which performs on state-of-the-art level in PPI extraction [21]) on IEPA and BioInfer corpora is presented in Table 6. All the numbers in Table 6 are averages taken over the 10-fold cross-validation. On the IEPA corpus, our method achieves a much higher performance than the graph kernel approach. But it should be noted that the IEPA corpus we used in previous experiments includes 336 positive instances and 308 negative instances while the IEPA corpus used in the experiments discussed in this section includes almost the same positive instances (335) and much more negative instances (482) (the one used in [21]) so that the result can be comparable with the graph kernel approach. On the BioInfer corpus, our method achieves the comparable performance with the graph kernel approach.

Tables 7 and 8 show the cross-corpus results measured with AUC and *F*-score respectively (in Table 8 we provide the optimal *F*-score results, choosing the positions from the precision/recall curves that would lead to highest *F*-scores). The graph kernel and our SVM model are both trained on IEPA (according to [21], the systems trained on IEPA can have high performance though the corpus is an order of magnitude smaller than AImed or BioInfer).

**Table 4**
Analysis of the BioPPISVMExtractor system recall error types.

| Error cause | Error number | Error proportion (%) |
|---|---|---|
| Pronoun resolution | 5 | 4.5 |
| Protein name recognition | 17 | 15.32 |
| Interaction extraction | 33 | 29.73 |
| No included in abstracts | 56 | 50.45 |
| Totals | 111 | 100 |

**Table 5**
Analysis of the BioPPISVMExtractor system precision error types.

| Error cause | Error number | Error proportion (%) |
|---|---|---|
| Pronoun resolution | 19 | 2.07 |
| Protein name recognition | 330 | 35.86 |
| Interaction extraction | 571 | 62.07 |
| Totals | 920 | 100 |

**Table 6**
Performance comparison with the graph kernel approach on IEPA and BioInfer corpora measured with precision, recall, *F*-score and AUC. The values are higher than those in Tables 2 and 3 since the performance is tested on clean data, i.e., the gene/protein names are pre-identified prior to the assessment.

| Corpus | Method | *P* | *R* | *F*-score (%) | AUC (%) |
|---|---|---|---|---|---|
| IEPA | Graph kernel | 69.6 | 82.7 | 75.1 | 85.1 |
| | SVM and rich feature sets | 72.9 | 90.3 | 80.6 | 87.9 |
| BioInfer | Graph kernel | 56.7 | 67.2 | 61.3 | 81.9 |
| | SVM and rich feature sets | 53.3 | 71.1 | 60.9 | 80.8 |

**Table 7**
Cross-corpus performance comparison with the graph kernel approach measured with AUC. The first column corresponds to the training corpus (IEPA); the second column corresponds to the methods used; the rest columns correspond to the results achieved on the test corpora (AImed, BioInfer, HPRD50 and LLL respectively).

| Corpus | Method | AUC (%) | | | |
|---|---|---|---|---|---|
| | | AImed | BioInfer | HPRD50 | LLL |
| IEPA | Graph kernel | 70.2 | 72.2 | 80.0 | 82.5 |
| IEPA | SVM and rich feature sets | 71.1 | 70.5 | 81.6 | 82.8 |

**Table 8**
Cross-corpus performance comparison with the graph kernel approach measured with *F*-score and optimal thresholds. The first column corresponds to the training corpus (IEPA); the second column corresponds to the methods used; the rest columns correspond to the results achieved on the test corpora (AImed, BioInfer, HPRD50 and LLL respectively).

| Corpus | Method | *F*-score (%) | | | |
|---|---|---|---|---|---|
| | | AImed | BioInfer | HPRD50 | LLL |
| IEPA | Graph kernel | 39.1 | 51.7 | 67.5 | 77.6 |
| IEPA | SVM and rich feature sets | 39.9 | 50.9 | 69.4 | 77.1 |

As can be seen from Tables 7 and 8, the results achieved with our method of SVM and rich feature sets can be comparable with those of the graph kernel approach which is state-of-the-art level. The advantage of the graph kernel approach lies in that it combines syntactic analysis with a representation of the linear order of the sentence. Similarly, our method achieves the comparable performance since it also combines the information of syntactic analysis (Link path and Link Grammar extraction result feature) and linear order of the sentence (word features, keyword feature, and protein names distance feature). This finding again verifies the conclusion that the combination of grammar engineering approaches and machine learning approaches may be a promising approach to achieve higher performance.

## 4. Conclusions and future work

The purpose of our study is to further boost the recall performance on DIP dataset while keeping an acceptable precision. In this paper we present a SVM-based PPI extraction system, BioPPISVMExtractor, which includes the whole procedure of PPI extraction from biomedical literature: pronoun resolution, protein name recognition and PPI extraction. In PPI extraction stage, besides several common features such as word features and keyword features, some new useful features including protein names distance feature, Link path feature and Link Grammar extraction result feature are introduced for SVM classification. Experimental evaluations of the BioPPISVMExtractor system with other PPI extraction systems tested on the DIP corpus indicate that our system can achieve substantially higher recall while still having a fairly good precision. In most cases, we speculate that, this is what the interaction database curators prefer since fewer interactions will be missed out. In future work, we plan to further explore the characteristics of SVM-based approach and refine our approach to achieve a better PPI extraction performance.

## Acknowledgments

This work is supported by grant from the Natural Science Foundation of China (No. 60373095 and 60673039) and the National High Tech Research and Development Plan of China (2006AA01Z151). The authors also wish to thank Dr. David Corney for sharing the evaluation datasets and results.

## Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.jbi.2009.08.013.

## References

[1] Zanzoni A, Montecchi-Palazzi L, Quondam M, Ausiello G, Helmer-Citterich M, Cesareni G. Mint: a molecular interaction database. FEBS Lett 2002;513(1):135–40.
[2] Bader G, Betel D, Hogue C. BIND – the biomolecular interaction network database. Nucleic Acids Res 2001;31(1):248–50.
[3] Bairoch A, Apweiler R. The swiss-prot protein sequence database and its supplement TrEMBL in 2000. Nucleic Acids Res 2000;28(1):45–8.
[4] Xenarios I, Fernandez E, Salwinski L, Duan XJ, Thompson MJ, Marcotte EM, et al. DIP: the database of interacting proteins. Nucleic Acids Res 2000;28(1):289–91.
[5] Blaschke C, Valencia A. The frame-based module of the Suiseki information extraction system. IEEE Intell Syst 2002;17(2):14–20.
[6] Ono T, Hishigaki H, Tanigam A, Takagi T. Automated extraction of information on protein–protein interactions from the biological literature. Bioinformatics 2001;17(2):155–61.
[7] Leroy G, Chen H, Martinez J, Eggers S, Falsey R, Kislin K, et al. Genescene: biomedical text and data mining. In: Proc. the third ACM/IEEE-CS joint conference on Digital libraries, Houston, TX, May 27–31; 2003.
[8] Corney DP, Buxton BF, Langdon WB, Jones DT. BioRAT: extracting biological information from full-length papers. Bioinformatics 2004;20(17):3206–13.
[9] Sekimizu T, Park HS, Tsujii J. Identifying the interaction between genes and gene products based on frequently seen verbs in MEDLINE abstracts. Genome Inform 1998;9:62–71.
[10] Pustejovsky J, Castano J, Zhang J, Kotecki M, Cochran B. Robust relational parsing over biomedical literature: extracting inhibit relations. In: Proc. the Pacific Symposium on Biocomputing, Hawaii, USA; 2002.
[11] Leroy G, Chen H, Martinez JD. A shallow parser based on closed-class words to capture relations in biomedical text. J Biomed Inform 2003;36(3):145–58.
[12] Temkin JM, Gilder MR. Extraction of protein interaction information from unstructured text using a context-free grammar. Bioinformatics 2003;19(16):2046–53.
[13] Rinaldi F, Schneider G, Kaljurand K, Dowdall J, Andronis C, Persidis A, et al. Mining relations in the GENIA corpus. In: Proc. Second European Workshop on Data Mining and Text Mining for Bioinformatics; 2004.
[14] Fundel K, Küffner R, Zimmer R. RelEx – relation extraction using dependency parse trees. Bioinformatics 2007;23(3):365–71.
[15] Ding J, Berleant D, Xu J, Fulmer AW. Extracting biochemical interactions from MEDLINE using a link grammar parser. In: Proc. the 15th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'03). IEEE Computer Society, Los Alamitos, CA, Nov 3–5; 2003.
[16] Ahmed ST, Chidambaram D, Davulcu H, Baral C. IntEx: a syntactic role driven protein–protein interaction extractor for bio-medical text. In: Proc. the ACL-ISMB Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics, Detroit, Michigan, June; 2005.
[17] Marcotte EM, Xenarios I, Eisenberg D. Mining literature for protein–protein interactions. Bioinformatics 2001;17(4):359–63.
[18] Donaldson I, Martin J, de Bruijn B, Wolting C, Lay V, et al. Pre-BIND and textomy-mining the biomedical literature for protein–protein interactions using a support vector machine. BMC Bioinform 2003;4(1):11.
[19] Xiao J, Su J, Zhou GD, Tan CL. Protein–protein interaction extraction: a supervised learning approach. In: Proc. the First International Symposium on Semantic Mining in Biomedicine, Hinxton, UK, April 10–13; 2005.
[20] Zhou D, He Y, Kwoh CK. Extracting protein–protein interactions from the literature using the Hidden Vector State Model. In: Proc. International workshop on bioinformatics research and applications, Reading, UK; 2006.
[21] Airola A, Pyysalo S, Björne J, Pahikkala T, Ginter F, Salakoski T. All-paths graph kernel for protein–protein interaction extraction with evaluation of cross-corpus learning. BMC Bioinform 2008;9(Suppl. 11):S2.
[22] Hunter L, Lu Z, Firby J, Baumgartner WA, Johnson HL, Ogren PV, et al. OpenDMAP: an open-source, ontology-driven concept analysis engine, with applications to capturing knowledge regarding protein transport, protein interactions and cell-specific gene expression. BMC Bioinform 2008;9:78.
[23] Bunescu R, Ge R, Kate R, Marcotte E, Mooney R, Ramani A, et al. Comparative experiments on learning information extractors for proteins and their interactions. Artif Intell Med 2005;33(2):139–55.
[24] Pyysalo S, GinterF Heimonen J, Björne J, Boberg J, Järvinen J, Salakoski T, et al. BMC Bioinform 2007;8:50.
[25] Fundel K, Kuffner R, Zimmer R. RelEx-Relation extraction using dependency parse trees. Bioinformatics 2007;23(3):365–71.
[26] Ding J, Berleant D, Nettleton D, Wurtele E. Mining MEDLINE: abstracts, sentences, or phrases? In: Proc. Pacific Symposium on Biocomputing, Hawaii, USA; 2002.
[27] Nédellec C. Learning language in logic–genic interaction extraction challenge. In: Proc. the 4th Learning Language in Logic Workshop, Bonn, Germany; 2005.
[28] Blaschke C, Valencia A. Can bibliographic pointers for known biological data be found automatically? Protein interactions as a case study. Comp Funct Genomics 2001;2(4):196–206.
[29] Yang ZH, Lin HF, Wu BD. BioPPIExtractor: a protein–protein interaction extraction system for biomedical literature. Expert Syst Appl 2009;36(2P1):2228–33.
[30] Yang ZH, Lin HF, Li YP. Exploiting the contextual cues for bio-entity name recognition in biomedical literature. J Biomed Inform 2008;41(4):580–7.
[31] Tsuruoka Y, Tateishi Y, Kim JD, Ohta T, McNaught J, Ananiadou S, et al. Developing a robust part-of-speech tagger for biomedical text. In: Proc. the 10th Panhellenic Conference on Informatics. Volos, Greece, Nov; 2005.
[32] Kim JD, Ohta T, Tateisi Y, Tsujii J. GENIA corpus – a semantically annotated corpus for bio-text mining. Bioinformatics 2003;19(Suppl. 1):i180–2.
[33] McDonald R, Pereira F. Identifying gene and protein mentions in text using conditional random fields. BMC Bioinform 2005;6(Suppl. 1):S6.
[34] Vapnik VN. The nature of statistical learning theory. NY: Springer-Verlag; 1995.
[35] Sugiyama K, Hatano K, Yoshikawa M, Uemura S. Extracting information on protein–protein interactions from biological literature based on machine learning approaches. Genome Inform 2003;14:699–700.
[36] Joachims T. Text categorization with support vector machines: learning with many relevant features. In: Proc. the European Conference on Machine Learning; 1998, p. 137–42.
[37] Joachims T. A statistical learning model of text classification with support vector machines. In: Proc.the 24th ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR); 2001, p. 128–36.
[38] Gimenez J, Marquez L. Fast and accurate Part-of-Speech tagging: the SVM approach revisited. In: Proc. the International Conference on Recent Advances in Natural Language Processing; 2003, p. 158–65.
[39] Keerthi S, DeCoste D. A modified finite Newton method for fast solution of large scale linear SVMs. J Mach Learn Res 2005;6:341–61.

[40] Joachims T. Making large-scale SVM learning practical. In: Advances in Kernel Methods – Support Vector Learning. Cambridge: MIT Press; 1999, p. 169–84 [Chapter 11].
[41] Sleator D, Temperley D. Parsing English with a Link Grammar. Carnegie Mellon University Computer Science technical report CMU-CS-91-196, Carnegie Mellon University; 1991.
[42] Grinberg D, Lafferty J, Sleator D. A robust parsing algorithm for LINK Grammars. In: Proc. the Second International Colloquium on Grammatical Inference and Applications, Lecture Notes in Artificial Intelligence. vol. 862, Springer-Verlag, p.78–92; 1994.
[43] Hanley JA, McNeil BJ. The meaning and use of the area under a receiver operating characteristic (roc) curve. Radiology 1982;143(1):29–36.
[44] Bradley AP. The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recognit 1997;30(7):1145–59.
[45] Buckland M, Gey F. The relationship between recall and precision. J Am Soc Info Sci 1994;45(1):12–9.
[46] Pyysalo S, Airola A, Heimonen J, Björne J, Ginter F, Salakoski T. Comparative analysis of five protein–protein interaction Corpora. BMC Bioinform 2008;9(Suppl. 3):S6.