Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 942435, 10 pages
http://dx.doi.org/10.1155/2015/942435

Research Article

A Novel Approach for Protein-Named Entity Recognition and Protein-Protein Interaction Extraction

Meijing Li, Tsendsuren Munkhdalai, Xiuming Yu, and Keun Ho Ryu

Database and Bioinformatics Laboratory, College of Electrical and Computer Engineering, Chungbuk National University, Cheongju 361-763, Republic of Korea

Correspondence should be addressed to Keun Ho Ryu; khryu@dblab.chungbuk.ac.kr

Received 27 February 2015; Revised 4 May 2015; Accepted 7 May 2015

Academic Editor: Nipon Theera-Umpon

Many researchers focus on developing protein-named entity recognition (Protein-NER) or PPI extraction systems. However, the studies about these two topics cannot be merged well; then existing PPI extraction systems' Protein-NER still needs to improve. In this paper, we developed the protein-protein interaction extraction system named PPIMiner based on Support Vector Machine (SVM) and parsing tree. PPIMiner consists of three main models: natural language processing (NLP) model, Protein-NER model, and PPI discovery model. The Protein-NER model, which is named ProNER, identifies the protein names based on two methods: dictionary-based method and machine learning-based method. ProNER is capable of identifying more proteins than dictionary-based Protein-NER model in other existing systems. The final discovered PPIs extracted via PPI discovery model are represented in detail because we showed the protein interaction types and the occurrence frequency through two different methods. In the experiments, the result shows that the performances achieved by our ProNER and PPI discovery model are better than other existing tools. PPIMiner applied this protein-named entity recognition approach and parsing tree based PPI extraction method to improve the performance of PPI extraction. We also provide an easy-to-use interface to access PPIs database and an online system for PPIs extraction and Protein-NER.

1. Introduction

Every year, a large number of biological experiments have been conducted, and many papers about proteins or genes have been published. With the amount and exponential increase of biology literature, it is almost impossible for biologists to keep up with the all the updated information in their research. Text mining technology that automatically extracts key information has been utilized in many biology fields. Until now, many systems based on special functions using text mining are published [1–3]. Text mining involves analyzing a large collection of documents in a manner that reveals specific information, such as the relationships and patterns buried in the collection, which is normally imperceptible to readers [4]. Two of the most important applications of text mining are bionamed entity recognition (Bio-NER) and mining correlations or associations such as protein-protein interactions (PPIs) from the literature [5].

Bionamed entity recognition is a very important task because it is directly related to extracting the PPIs. If we cannot identify bionamed entities correctly, then we cannot extract the correct PPIs. Our goals are the number of recognized bionamed entities and accuracy of recognition. Thus, the main issues in this topic are as follows: how to increase the recognition accuracy and how to identify as many proteins and genes as possible. However, until now, it is difficult to satisfy the two questions together.

Three basic approaches are usually applied: the dictionary-based approach, rule-based approach, and machine learning-based approach. The dictionary-based approach refers to the use of word lists or databases containing protein or gene names, for comparison and identification. The rule-based approach is used, for example, on extracting events based on the cooccurrence of strings, prefixes, or syntactic tags. Machine learning-based approach is a method which applies machine learning algorithm to classify bionamed entities from sentences. Comparing with above two approaches, machine learning-based approach is the most popular topic because biology named entities are usually not represented by recommended names in the raw text data.

Many bionamed entity recognition systems and methods have been proposed. The applied machine learning algorithms are HMM [6, 7], SVM [8, 9], MEMM [10], CRFs [11–13], and so forth. To increase the accuracy of recognition, several researches applied two machine learning algorithms together [9, 14].

The development of PPI extraction system is more popular than Protein-NER systems. Many PPI extraction systems have been published too, which have specific advantages, for example, PPI Finder [5], CBioC [25], and iHOP [26], among others. PPI Finder focuses on extracting human PPIs and gives an interface for searching the extracted PPIs. It also constructs a database to store the PPIs and related information. CBioC extracts binary relationships between biological entities automatically from the biomedical literature and provides a platform that allows community collaboration in the annotation of the extracted relationships. iHOP is an online service that provides gene-guided network as a natural way of accessing PubMed abstracts and brings all the advantages of the internet to scientific literature research. However, these systems have minor consideration in improving Protein-NER methodologies.

From 2004, the BioCreAtIvE challenge began which is held by BioCreAtIvE (A critical assessment of text mining methods in molecular biology). Three main tasks were posed at the first BioCreAtIvE challenge: the entity extraction task, the gene name normalization task, and the functional annotation of gene products task. Until now, five BioCreative Challenge Evaluations and Workshops (BioCreative I, II, II.5, III, and IV) were held [28, 29]. The last workshop contain three tasks: (I-Triage) a collaborative biocuration-text mining development task for document prioritization for curation; (II-Workflow) a biocuration workflow survey and analysis task; and (III-Interactive TM) an interactive text mining and user evaluation task.

Many machine learning-based Protein-NER methods are proposed. However, the most published tools for PPI extraction did not apply new machine learning-based Protein-NER approach (e.g., iHOP [26], PPI Finder [5], and PPInterFinder [24]). These studies still prefer to use dictionary-based Protein-NER method because machine learning-based Protein-NER is more time-consuming and the performance is not enough high [5, 25]. But actually, if you only use dictionary-based Protein-NER method to identify the proteins in the text, we will miss so many protein names which not are included in the "dictionary." Furthermore, most published studies on PPI extraction [18, 21, 23, 24] do not consider the Protein-NER step and just developed methods for PPIs extraction. In other words, these two related topics are researched independently. So it is difficult to merge and be applied by users.

In the study on the extraction of PPI patterns, the step of Protein-NER is important [30]. If the most of protein-named entities we found are not correct or many protein-named entities are not identified by Protein_NER method, the PPI extraction task cannot be continued. It is the first step of all biomedical information extraction tasks including PPI extraction so that it directly affects the quality of results [30]. For example, if we just find out half of the protein-named entities, then our result will just contain less than half of the information of PPIs. So in our study, we constructed a high quality Protein-NER model and merged into PPI process. Furthermore, it used two different methods, so that it not only can identify the protein names that are included in protein dictionary, it can also identify the proteins which are not included in protein dictionary. Combining this Protein-NER method to our PPI extraction process based on passing tree can efficiently improve the extraction of PPI patterns, because this approach will save time to find protein names again in the step of PPI extraction.

Many statistic methods have been used to find PPIs [31, 32]. They often find two proteins' occurrence frequency in a sentence or some mathematic formula to calculate the correlations. For example, if two proteins appear together in a sentence or in a paragraph frequently, the two proteins interact with each other. However it cannot achieve the more accurate interactions between proteins, because two proteins can have many other relationships except interaction. Text feature (interaction word) based PPI extraction methods are frequently applied to discover the protein interaction pattern because this method can discover clearer interactions between the proteins. But if we need to check all the sentences, it will be time-consuming. Therefore, we propose our approach combining these two methods in discovering weighted PPIs by occurrence frequency. Comparing with text feature based PPI exaction methods, this approach can improve the efficiency. Comparing with statistic methods, it also improves the performance. Currently, most of these researches seldom show the information of PPI types (e.g., bind, link, and inhibit); the type of PPI is very important information for biology researchers. In our approach, we also improve it by including information about protein interaction type referred in the literature.

2. Methods

The proposed PPI extraction system is constructed via three main models: the natural language processing (NLP) model, protein-named entity recognition model, and protein interaction discovery model. The workflow of PPIMiner is as shown in Figure 1.

2.1. Data Preprocessing: Natural Language Processing. As known, NLP is the first step of text mining. In this study, the NLP model mainly contains two parts: sentence parsing and part-of-speech (POS) tagging. Here, Penn Treebank English POS tag set [33] is used for part-of-speech (POS) tag because this POS tag set is a standard tag set. The PPIMiner contains a Protein-NER model to identify the protein name, so we do not need special biological POS tag set for biology text. We applied open source of Stanford Log-Linear Part-Of-Speech Tagger which is based on Penn Treebank English POS tag set for our NLP model. The GENIA corpus 3.2 [34] is used as training dataset of protein name recognition model, and the format of dataset is an XML file. So, it is easy to classify the protein names from the text.

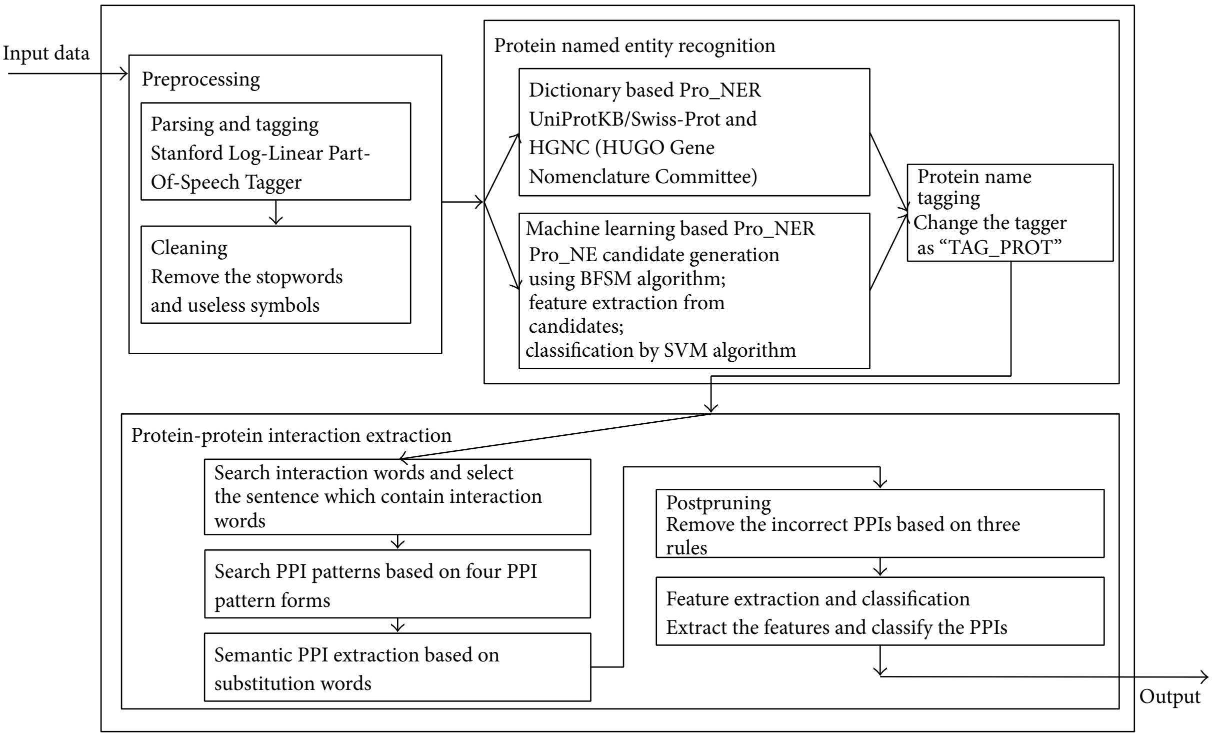

Input data
Preprocessing
Parsing and tagging
Stanford Log-Linear Part-Of-Speech Tagger
Cleaning
Remove the stopwords and useless symbols
Protein named entity recognition
Dictionary based Pro_NER
UniProtKB/Swiss-Prot and HGNC (HUGO Gene Nomenclature Committee)
Machine learning based Pro_NER
Pro_NE candidate generation using BFSM algorithm;
feature extraction from candidates;
classification by SVM algorithm
Protein name tagging
Change the tagger as "TAG_PROT"
Protein-protein interaction extraction
Search interaction words and select the sentence which contain interaction words
Search PPI patterns based on four PPI pattern forms
Semantic PPI extraction based on substitution words
Postpruning
Remove the incorrect PPIs based on three rules
Feature extraction and classification
Extract the features and classify the PPIs
Output

Figure 1: Workflow of PPIMiner. This work used two datasets, GENIA and PubMed. The extracted PPIs are stored into our PPI database.

2.2. ProNER: Protein-Named Entity Recognition. Studies on PPI exaction systems mainly used the dictionary-based Protein-NER method to search the proteins. In order to increase the number of proteins extracted, we used two methods, dictionary-based Protein-NER method and machine learning-based Protein-NER method, to construct the protein-named recognition model, which has different advantages, respectively. User interface is as Figures 2(a) and 2(b).

2.2.1. Dictionary-Based Protein-NER. The result of dictionary-based Protein-NER method is the most believable result, comparing with other methods'. And it is easier to implement than others. Almost published systems used this method [5, 25]. In our study, we also used this method. UniProtKB/Swiss-Prot Database [35] is used as PPIMiner's protein dictionary. There are 200,839 protein entries in the database. We select the 19,304 human proteins to use. HGNC (HUGO Gene Nomenclature Committee) [36] is used as a gene dictionary to discover the PPI represented by gene names. 5,038 extracted gene entities are restored in our database. To search as many protein-named entities as possible, we did not use only recommendation name of protein or gene; we also used other names in protein or gene entity database.

2.2.2. Machine Learning-Based Protein-NER. This Protein-NER method contains three steps: candidate generation (boundary detection), for feature extraction and classification. The reason we use this method is that it can find novel protein and gene names, which are not included in the dictionaries or the proteins and genes that are not represented by standard names. So, we used the machine learning-based method to search the more bionamed entities.

Candidate Generation Using the BFSM Algorithm. Boundary detection is a difficult task for machine learning-based methods. If we analyze more than 100,000 papers, the text dataset is also too large to identify using data mining algorithms. In this paper, to solve this problem, we used Bayesian probability based Finite State Machine (BFSM) to detect the boundary of bionamed entities and generate the candidates for classification. BFSM algorithm is proposed by us in 2011 [27]. And it is proved that this algorithm is suitable to apply for Protein-NER. Here, we use this algorithm to extract the rules of POS set of protein names from text data. And the corpora which satisfy the rules of POS set will be generated as protein-named entity candidates.

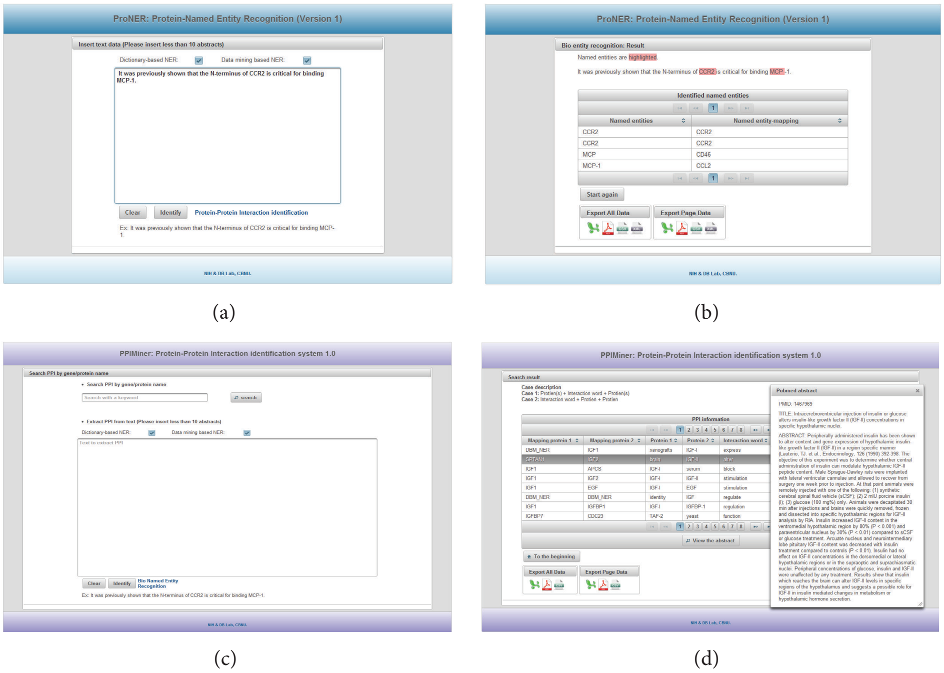
The process is as follows.

(i) Mine the frequent POS patterns of bionamed entities using Apriori algorithm.

(ii) Calculate the confidence and search the association rules in the k-frequent itemsets ($k \geq 1$) that satisfy the minimum confidence.

(iii) Construct the BFSM model according to the frequent patterns and confidence values between the items.

(iv) Using the BFSM model, generate the candidates of protein-named entities.



ProNER: Protein-Named Entity Recognition (Version 1)

Insert text data (Please insert less than 10 abstracts)

Dictionary-based NER: ☑ Data mining based NER: ☑

It was previously shown that the N-terminus of CCR2 is critical for binding MCP-1.

Clear | Identify | Protein-Protein Interaction identification

Ex: It was previously shown that the N-terminus of CCR2 is critical for binding MCP-1.

NIH & DB Lab, CBNU.

(a)

ProNER: Protein-Named Entity Recognition (Version 1)

Bio entity recognition: Result

Named entities are highlighted.

It was previously shown that the N-terminus of CCR2 is critical for binding MCP -1.

Identified named entities

Named entities	Named entity-mapping
CCR2	CCR2
CCR2	CCR2
MCP	CD46
MCP-1	CCL2

Start again

Export All Data | Export Page Data

NIH & DB Lab, CBNU.

(b)

PPIMiner: Protein-Protein Interaction identification system 1.0

Search PPI by gene/protein name

- Search PPI by gene/protein name

Search with a keyword | search

- Extract PPI from text (Please insert less than 10 abstracts)

Dictionary-based NER: ☑ Data mining based NER: ☑

Text to extract PPI

Clear | Identify | Bio Named Entity Recognition

Ex: It was previously shown that the N-terminus of CCR2 is critical for binding MCP-1.

NIH & DB Lab, CBNU.

(c)

PPIMiner: Protein-Protein Interaction identification system 1.0

Search result

Case description
Case 1: Protien(s) + Interaction word + Protien(s)
Case 2: Interaction word + Protien + Protien

PPI information

Mapping protein 1	Mapping protein 2	Protein 1	Protein 2	Interaction word
DBM_NER	IGF1	xenografts	IGF-I	express
SPTAN1	IGF2	brain	IGF-II	alter
IGF1	APCS	IGF-I	serum	block
IGF1	IGF2	IGF-I	IGF-II	stimulation
IGF1	EGF	IGF-I	EGF	stimulation
DBM_NER	DBM_NER	identity	IGF	regulate
IGF1	IGFBP1	IGF-I	IGFBP-1	regulation
IGFBP7	CDC23	TAF-2	yeast	function

View the abstract

To the beginning

Export All Data | Export Page Data

Pubmed abstract

PMID: 1467969

TITLE: Intracerebroventricular injection of insulin or glucose alters insulin-like growth factor II (IGF-II) concentrations in specific hypothalamic nuclei.

ABSTRACT: Peripherally administered insulin has been shown to alter content and gene expression of hypothalamic insulin-like growth factor II (IGF-II) in a region specific manner (Lauterio, TJ. et al., Endocrinology, 126 (1990) 392-398. The objective of this experiment was to determine whether central administration of insulin can modulate hypothalamic IGF-II peptide content. Male Sprague-Dawley rats were implanted with lateral ventricular cannulae and allowed to recover from surgery one week prior to injection. At that point animals were remotely injected with one of the following: (1) synthetic cerebral spinal fluid vehicle (sCSF); (2) 2 mIU porcine insulin (I); (3) glucose (100 mg%) only. Animals were decapitated 30 min after injections and brains were quickly removed, frozen and dissected into specific hypothalamic regions for IGF-II analysis by RIA. Insulin increased IGF-II content in the ventromedial hypothalamic region by 80% (P < 0.001) and paraventricular nucleus by 30% (P < 0.01) compared to sCSF or glucose treatment. Arcuate nucleus and neurointermediary lobe pituitary IGF-II content was decreased with insulin treatment compared to controls (P < 0.01). Insulin had no effect on IGF-II concentrations in the dorsomedial or lateral hypothalamic regions or in the supraoptic and suprachiasmatic nuclei. Peripheral concentrations of glucose, insulin and IGF-II were unaffected by any treatment. Results show that insulin which reaches the brain can alter IGF-II levels in specific regions of the hypothalamus and suggests a possible role for IGF-II in insulin mediated changes in metabolism or hypothalamic hormone secretion.

NIH & DB Lab, CBNU.

(d)

Figure 2: User interface of the proposed ProNER model and PPI extraction model. (a) Interface of ProNER for biomedical sentence input. Here, users can select the Protein-NER methods, dictionary-based method, or data mining-based method. Users execute the PPIMiner by using the button "Identify." (b) Final result of Protein-NER. Here, the "named entity-mapping" represents the recommendation name of this protein or gene name or other named entities that were extracted using data mining-based (machine learning-based) method. (c) Interface of PPI extraction system for inputting biomedical sentence. User can also select the Protein-NER method. User can search the protein-protein interactions from our PPI database. (d) PPI patterns searched by proposed model. From the result page, we can get the information of PPIs, the type of PPIs, PubMed ID of resource documents, and the full abstract of the resource documents.

To understand BFSM easily, an example of finite state network to generate POS rules and generated POS rules is given as in Figure 3.

Feature Selection and Extraction. Until now, many features for Protein-NER are generated [11]. However, the target bionamed entities are different; the suitable features are different too. So, according to the external rules of protein and gene molecule name, we set up several distinguishing features for protein and gene named entity recognition. We extract these features from the candidates to converted candidates list into a tabular dataset.

In this study, our extracted features as follows.

(1) Sent_Uppercase: the first letter of a word is an uppercase letter, which is not located in the middle of sentence.

(2) Word_Uppercase: uppercase letter is located in the middle of a word.

(3) Word_Symbol: a word contains comma (,), hyphen (-), and slash (/).

(4) Word_Num: numbers are located in the middle of a word.

(5) Word_Alphabet: special letters are located in the middle of a word.

(6) Biology_Word: the protein-named entity candidate contains special words: protein, gene, receptor, and factor.

(7) Length: the length of the words.

(8) Suffix_Word: suffix word of protein-named entity candidate.

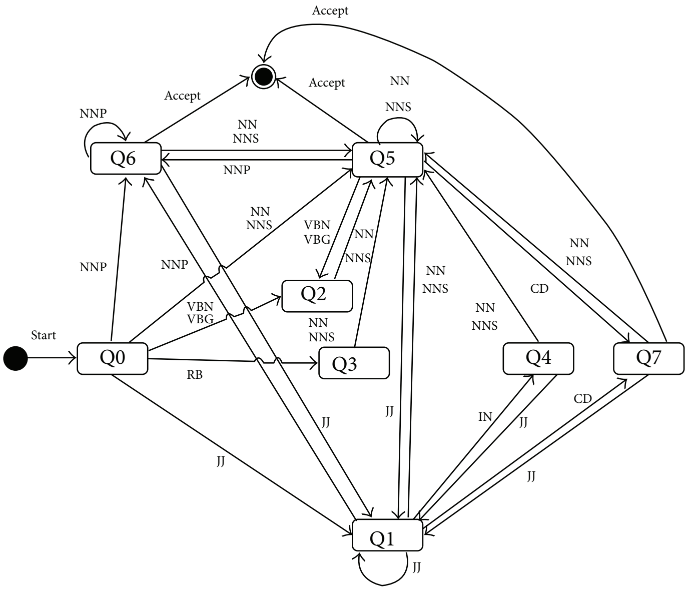
(9) Prefix_Word: prefix word of protein-named entity candidate.

(10) Suffix_Letter: suffix letter of protein-named entity candidate.

(11) Prefix_Letter: prefix letter of protein-named entity candidate.

(12) POS: POS set of protein-named entity candidate.

Classification. We use the SVM algorithm [37] to classify the protein name based on these features from the papers. Compared with other machine learning methods, SVM algorithm is suitable for our text dataset and the accuracy is higher than others. When we begin to test, this kind of method is more time-consuming than other classification algorithms. But the accuracy is more important than running time for us because our purpose is to find more correct protein-named entities using this system. From the result of



(1) JJ = TRUE 17055 ⇢ NN = TRUE 15723 conf: (0.92)

(2) VBN = TRUE 746 ⇢ NN = TRUE 680 conf: (0.91)

(3) RB = TRUE 543 ⇢ NN = TRUE 476 conf: (0.88)

(4) VBG = TRUE 868 ⇢ NN = TRUE 704 conf: (0.81)

(5) NNP = TRUE 9342 ⇢ NN = TRUE 7021 conf: (0.75)

(6) IN = TRUE 550 ⇢ NN = TRUE 360 conf: (0.65)

(7) CD = TRUE 4565 ⇢ NN = TRUE 2891 conf: (0.63)

(8) NN = TRUE 30266 ⇢ JJ = TRUE 15723 conf: (0.52)

(9) NNP = TRUE 9342 ⇢ JJ = TRUE 3306 conf: (0.35)

(10) NNS = TRUE 1838 ⇢ JJ = TRUE 617 conf: (0.34)

(11) NNS = TRUE 1838 ⇢ NN = TRUE 605 conf: (0.33)

(12) NN = TRUE 30266 ⇢ NNP = TRUE 7021 conf: (0.23)

(13) CD = TRUE 4565 ⇢ JJ = TRUE 967 conf: (0.21)

(14) JJ = TRUE 17055 ⇢ NNP = TRUE 3306 conf: (0.19)

(15) CD = TRUE 4565 ⇢ NNP = TRUE 740 conf: (0.16)

Figure 3: An example of constructed finite state network and generated POS rules. These rules are extracted form GENIA data. It refers from [27].

experiment, we know that the performance of classification algorithm is better, when the training data is enough. The SVM algorithm we used is libSVM in the experiments.

2.3. Protein-Protein Interaction Discovery. Based on these identified protein/gene names, we used two kinds of methods, the interaction pattern-based method and frequent pattern-based method, to discover the PPI patterns. The interaction pattern-based method is that we discover the interaction patterns (interaction words) between two proteins such as "active" and "bind." We used the parsing tree based interaction pattern scanning approach to discover the PPI patterns. To search the patterns easily, the parsing tree is used to search the algorithm structure in this research, and we then search the surround protein names to discover the correct PPIs. The frequent pattern-based method searches for the pattern that frequently occurs in the whole dataset. The frequency is the weight of the PPI patterns exacted from the interaction pattern-based method. The final discovered PPI patterns include the interaction weight and represent the interactions between proteins more accurately than the result of just using one method, interaction patterns based or frequent patterns based method (Figures 2(c) and 2(d)).

We use a parsing tree to search the interaction patterns, and this method is much suitable for the PPIMiner that includes Protein-NER model because when we search the protein-named entity candidates, we change the POS set to protein or gene mark "TAG_PROT." It can decrease the deep of parsing tree and then decrease the complex.

In Figure 4, a simple example shows how to change the POS set to protein or gene mark "TAG_PROT."

2.3.1. PPI Candidate Extraction. In the previous studies [23, 24, 38–40], they discussed how to describe relation of two bionamed entities by expressed forms. Here we also used these forms.

The process of PPI extraction is as follows.

(i) Search the interaction patterns in a sentence.

(ii) Discover the interaction patterns in the sentence that includes the cooccurring genes or proteins.

Form 1: protein(s) + interaction word (verb/noun) + protein(s).

Form 2: interaction word (noun) + protein + protein.

Form 3: protein(s) + (null, "/," ":," "and") + protein(s) + interaction word.

(iii) Map the recognized protein-named entity name onto gene and recommend gene name in dictionary.

(iv) Postpruning based on prepositions to filter out incorrect PPIs based.

Here, the 113 interaction patterns are getting from [41] and 175 interaction words are getting from HPDR50 [42]. The total interaction words are 1,224 words that contained derivatives of these words and deleted duplicated works. To extract more patterns, we derived the interaction patterns to other kinds of POS interaction patterns.

2.3.2. Semantic PPI Information Extraction. For discovery of more semantic PPIs, substitution words are discovered to find out more PPIs which are explained by two or more sentences. The substitution word list includes "this protein," "that protein," "the protein(s)," "these proteins," and "those proteins." And we will check the subject in the previous sentence. If the subject is not protein name, the PPI will be skipped.

Protein-named entities are found out to discover PPIs firstly when researchers use statistical methods. And it is

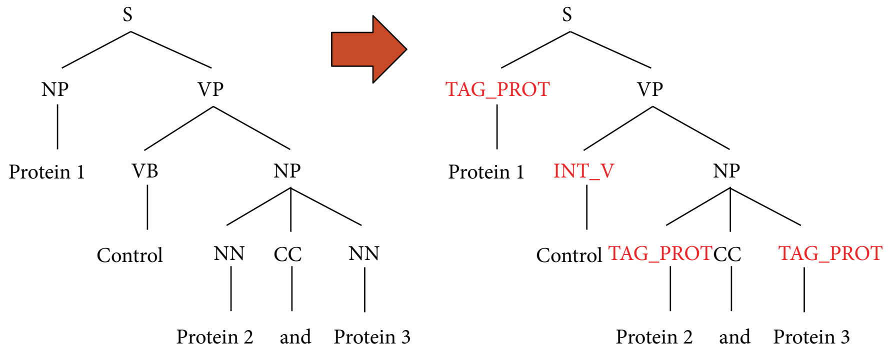


FIGURE 4: A simple example about the replacement POS set to protein or gene mark "TAG_PROT."

not important which one is extracted firstly between the interaction word and protein-named entities when feature based methods or pattern-based methods are used to discover PPIs. In our research, we search PPI interaction words firstly to increase search efficiency. If we search protein-named entities firstly and decide to analyze the sentence deeply, it is very time-consuming because too much substation words exist in a paper.

2.3.3. Pruning of PPIs. To improve the accuracy of prediction, a pruning method is proposed. Two kinds of conditions need to be satisfied.

Pruning 1. If the words "by," "in," "of," and "as" are in front of the protein name, the extracted PPIs are pruned.

It can filter out incorrect PPIs which are only considered the combinations of protein and interaction words.

Pruning 2. If a negative word exists in the sentence, the extracted PPIs are pruned.

Form 1: "not" + interaction word.

Form 2: "no" + protein(s) + interaction words.

Form 3: "neither" protein "nor" protein + interaction word.

In Form 3, if other negative word exists, the extracted PPIs are not pruned (e.g., neither Protein 1 nor Protein 2 interacts with Protein 3).

The purpose of pruning method is to solve the low value of recall problem in many cases.

2.3.4. Feature Extraction and Classification. To extract more correct PPIs, several features are generated which is suitable to classify the PPI. Three classification methods (SVM, C4.5, and neural network) are used to compare the result. The classification algorithms are provided by the Weka.

3. Results and Discussion

3.1. Dataset

3.1.1. Dataset for Protein-NER Evaluation. In this research, we used three datasets extracted from the GENIA. GENIA corpus is the largest corpus of its type currently available, comprising 2000 abstracts with 18,545 sentences containing 39,373 named entities, and it is usually used to train and test the Protein-NER model. After sentence parsing and POS tagging, we converted the GENIA dataset into table dataset which includes 305,546 records, and 83,403 records are protein-named entities out of the all dataset (shown in Tables 1 and 2).

3.1.2. Dataset for PPI Extraction Evaluation. In this research, we used five corpora as experiment dataset which are provided by [42]: AIMed, BioInfer, HpDR50, IEPA, and LLL (http://mars.cs.utu.fi/PPICorpora/GraphKernel.html). These corpora have unified format. The most of researches on PPI extraction were used these dataset for evaluation (shown in Table 3).

We collect 1,571,293 paper abstracts that are related to gene and protein from PubMed database. We save the paper abstracts to our database to extract PPIs. Finally, we compare the PPI patterns with HPRD PPIs to analyze the result. The HPRD contains 39,194 PPI extracted from 20,071 papers in PubMed. This file contains human protein-protein interactions in a tab delimited format.

3.2. Evaluation Criterion. Precision determines the fraction of records that actually turn out to be positive in the group the classifier has declared as a positive class:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{1}$$

The recall measures the fraction of positive examples correctly predicted by the classifier. It represents the ability to find protein-named Entities (Protein-NEs) in the dataset:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{2}$$

The F-measure represents a harmonic-mean between precision and recall and takes both measures into account. A high value of F-measure ensures that both of precision and recall are reasonably high:

$$F\text{-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{3}$$

Table 1: Statistic on GENIA corpus.

Corpus	Number of abstracts	Number of sentences	Number of named entities
GENIA corpus	2,000	18,545	39,373

Table 2: Statistic on converted GENIA corpus.

Corpus	Number of records	Number of Bio-NEs	Number of attributes
Converted GENIA corpus	305,546	83,403	12

Table 3: Statistics on five corpora.

Corpus	AIMed	BioInfer	HPRD50	IEPA	LLL
Positive pairs	1000	2534	163	335	164
Negative pairs	4834	7132	270	482	166
All pairs	5834	9666	433	817	330
Sentences	1955	1100	145	486	77

3.3. Performance of Protein-NER. GENIA corpus data is classified by 36 class labels. In this paper, we just select data which is included in "protein-molecule" class label as protein-named entities because GENIA corpus data is used as training data for constructing the protein-named entity recognition model. Although there are 7 class labels related to protein and 5 class labels related to DNA, they did not include the external features of protein or gene names. For example, the data which are included in class "protein-family-or-group" and "DNA-family-or-group" cannot describe the protein or gene entity's names; the data are just the protein or gene categories such as "receptor" or "gene."

All the experiments are carried out on a unified computer platform with a 2.93 GHz CPU and a 4 GB RAM.

3.3.1. Effect of Prepruning by BFSM. In our Protein-NER model, prepruning directly affects the accuracy of protein identification. In order to know how to influence the identification of protein-named entities, we did two experiments. In the first experiment, at first we converted GENIA text dataset into table dataset which contain 12 features for training the classifier. The two class labels of converted dataset are "protein-named entity" and "Not_Protein-Named Entity." The number of instance in dataset is 305,546. We use the Support Vector Machine (SVM) algorithm in Weka 3.6 as classifier. The performance evaluation result is very high, although we just select 5% dataset. Table 4 shows a direct proportion between sample size and F-measure. But when we use the text dataset, the accuracy is difficult to be higher than 95% because those boundary detection methods are very difficult to find out the boundary of protein-named entity. In previous research, they use sliding window technique to generate the entity candidates [8]. 305,547 candidates were generated using our method. The original text dataset includes 402,745 words. So, comparing with the number of candidates generated by sliding window, the number of candidates generated by BFSM is much small.

Table 4: Performance evaluation using prepruned dataset by size of sample dataset.

Sample size	5%	10%	20%	40%	60%
Precision	97.7	98.4	98.9	99.4	99.7
Recall	97.6	98.3	98.9	99.4	99.7
F-measure	97.4	98.2	98.9	94.4	99.7
Accuracy	97.6	98.3	98.9	99.4	99.6

3.3.2. Effect of Features on Classification. We search and detect features to classify the protein or gene named entities, and we analyze the features' effect on classification. Table 5 shows the distributions of 6 binary features in two classes, respectively. From Table 5, we know that the percentage of two kinds of values in two different class labels is much different. The percentage of value "True" of Features "Word_Symbol" and "Word_Alphabet" in class "protein-named entity" is much higher than in "Not_Protein-named entity." To get more clear values to evaluate the features, we calculate the Gain Ratio of the features by Weka 3.6 as shown in Table 5. The Gain Ratio is just the ratio between the information gain and the intrinsic value. Here, the information gain is equal to the total entropy for an attribute if a unique classification can be made for the result attribute of each of the attribute values. The information gain for an attribute $a \in \mathrm{Attr}$ is defined as follows:

$$\mathrm{IG}(Ex,a) = H(Ex) - \sum_{v\in\mathrm{values}(a)} \frac{|\{x \in Ex \mid \mathrm{value}(x,a) = v\}|}{|Ex|} \cdot H(\{x \in Ex \mid \mathrm{value}(x,a) = v\}), \tag{4}$$

where Attr is a set of all attributes and Ex a set of all training examples, value (x, a) with $x \in Ex$ fines the value of a specific example x for attribute $a \in$ Attr, H specifies the entropy. From this experiment, we found that the most informative features are "Sent_Uppercase," "Word_Num," and "Word_Symbol." Furthermore, the performance evaluation values are close to 100%. It means that the features are very suitable for protein-named entity recognition. We did other feature analysis experiments about DNA or others. But the feature ranking is different according to the class target.

3.3.3. Comparison among Submodels in PPIMiner. To know the performance of submodels, we did the experiment to recognize the protein-named entities using two submodels, respectively, as in Table 6. From the result, we can get that the precision of dictionary-based model is much higher than others, but recall F-measure and accuracy is too lower than others. And when we use the two submodels together, the F-measure is the best. Although many machine learning-based Protein-NER methods are proposed, the PPI extraction systems prefer to use dictionary-based approach than machine learning-based Bio-NER approach to extract the correct PPI patterns. However too many protein-named entities are missed, if we just use dictionary-based approach, and too many PPI patterns are missed.

Table 5: Gain Ratio and ranking of the features and distribution of entities by features.

Feature	Gain Ratio	Ranking	Pro_NE (%)	Not Pro_NE (%)	Pro_NE/Not_Pro_NE
Sent_Uppercase	0.08827	1	24.3	15.7	1.55
Word_Num	0.0839	2	10.5	17.8	0.59
Word_Symbol	0.05184	3	37.3	16.7	2.23
Suffix_Letter	0.04518	4	—	—	—
Word_Uppercase	0.04284	5	24.5	15.5	1.58
Suffix_Word	0.0388	6	—	—	—
Prefix_Word	0.03851	7	—	—	—
Prefix_Letter	0.03254	8	—	—	—
Length	0.02472	9	—	—	—
POS	0.02108	10	—	—	—
Word_Alphabet	0.01411	11	22.5	5.5	4.09
Biology_Word	0.00833	12	21.8	7.9	2.76

Table 6: Performance comparison with Protein-NER model.

Method	Precision (%)	Recall (%)	F-score (%)
Dictionary-based Protein-NER	97.15	16.89	28.77
Machine learning-based Protein-NER	92.47	87.13	89.72
Dictionary and machine learning-based Protein-NER	92.47	89.17	90.97

3.3.4. Comparison with Other Existing Methods. To test the performance of our Protein-NER model, we did the comparison experiment with three published systems and two resent published researches. We used the 3-fold cross-validation method to evaluate. We also used the same training dataset to construct the identification models and used same test dataset to evaluate. And the performance results of the two resent published researches about Bio-NER referred to published papers. From the result in Table 7, we can get that our Protein-NER model has the highest F-measure. Here, ABNER and Penn BioTagger are applied conditional random fields to recognize the protein-named entities. And the result is better than others. Conditional random fields algorithm is based on conditional probability to calculate the possibility of protein-named entities. In our approach, BFSM we used is also based on conditional probability to generate the rules of POS for protein-named entities. Finally, we applied vector based methods to candidates so that more properties of protein-named entities are discovered.

3.4. Performance of PPI Extraction

3.4.1. Result of PPI Extraction. We use this system to extract 40,466 PPIs from 477,008 abstracts in PubMed. We extract 6,072 patterns from 20,071 papers which are included in HPRD. We did not extract the PPIs as many as manual extraction in HPRD. But the number of PPI patterns will continue to increase. And we extracted more than 7,919 rules from other abstracts. For example, we extracted 47 PPI patterns related to protein “IGF1.” In our PPIs database, several extracted PPIs maybe have incorrect PPIs. But user can check by himself from the original text provided by PPIMiner. The system is freely accessible at http://210.115.182.155:8080/PPIMiner/. We purpose extracting more possible PPIs. And our system also shows the type of relation between interacting proteins. Actually, it is more informative for biologists that the proteins are how to interact.

3.4.2. Compare with Other PPI Extraction Methods. To compare with existing methods, the standard dataset, AIMed corpus, was used. So we extract the result from the papers. As in Table 8, our proposed method has higher performance.

From the result, we can get that feature or pattern-based methods’ performance is better than subsequence kernels or all-path graph kernel. However, it is important to find out right patterns or features of PPI information. If merged all the discovered patterns and features, the performance will be improved.

4. Conclusion

Many human PPI databases (HPRD, BioGRID, and BIND) published extract PPIs manually. But it is impossible to extract all the PPIs manually from the articles because the number of biomedical articles is growing at a very fast rate. So we need to automatically extract PPIs and upgrade the PPI database regularly.

In this paper, we developed an online system of the protein-protein interaction extraction based on SVM and parsing tree, PPIMiner. The PPI extraction system is constructed via the natural language processing model, protein-named entity recognition model, and protein-protein interaction discovery model. We used the two methods, dictionary-based method and machine learning-based method, to construct the protein name recognition models which have a different advantage, respectively. We extract the PPI patterns based on these identified protein-named entities using parsing tree. Until now, our database stored 40,466 PPIs.

Table 7: Performance comparison with other published Bio-NER systems.

System	Precision (%)	Recall (%)	F-score (%)
ABNER [15]	88.1	84.8	86.41
Penn BioTagger [16]	87.3	85	86.13
Sun et al. (2007) [12]	69.03	78.05	73.27
Li et al. (2009) [14]	71.14	81.63	76.02
BANNER (2008) [17]	89.3	85.06	87.13
ProNER-machine learning based	92.47	87.13	89.72
ProNER	92.47	89.17	90.79

Table 8: Comparison with other methods.

System	Description	F-score (%)
Saetre et al. [18]	Feature-based	64.2
Miwa et al. [19]	Multiple kernels	60.8
Kim et al. [20]	Walk-weighted subsequence kernels	56.6
Airola et al. [21]	All-path graph kernel	56.4
Niu et al. [22]	All-path graph kernel	53.5
Bui et al. [23]	RBF kernel	61.2
PPInterFinder [24]	Pattern matching	66.05
PPIMiner	Pattern matching, feature-based	66.8

The advantages of our developed system are as follows. First, in our PPI extraction system, we merged protein-named entity recognition model we proposed to PPI extraction model. So, comparing with the systems which just use the dictionary-based protein-named entity recognition method, we can extract more PPIs. The accuracy of protein-named entity recognition model is higher than other existing models and published methods. Second, PPIMiner provides three main functions: PPIs search in database, protein-named entity recognition, and PPI extraction from text data. We made an interface for protein-named entity recognition model. From PPIMiner interface, users are easy to access our protein-named entity recognition model and users can apply this model for their research.

In addition, our method used 12 features for construction of protein-named entity recognition model. And from the accuracy of classification result, we can know that these features are much suitable for protein-named entity recognition.

For feature work, we plan to supplement the protein and gene dictionary to increase the accuracy of identification. We also plan to develop the PPI extraction algorithm to extract more PPIs. The extracted PPIs in our database will continue to be updated.

Conflict of Interests

The authors declare that there is no conflict of interests.

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (no. 2013R1A2A2A01068923).

References

[1] H. Wang, Y. Ding, J. Tang et al., “Finding complex biological relationships in recent PubMed articles using Bio-LDA,” *PLoS ONE*, vol. 6, no. 3, Article ID e17243, 2011.

[2] S. Anthony, V. Sintchenko, and E. Coiera, “Text mining for discovery of hostpathogen interactions,” *Infectious Disease Informatics*, vol. 1, pp. 149–165, 2010.

[3] A. Abi-Haidar and L. M. Rocha, “Collective classification of biomedical articles using T-cell cross-regulation,” in *Proceedings of the Artificial Life XII: 12th International Conference on the Simulation and Synthesis of Living Systems*, H. Fellermann, M. Dörr, M. Hanczyc et al., Eds., pp. 706–713, 2010.

[4] H.-J. Dai, Y.-C. Chang, R. T.-H. Tsai, and W.-L. Hsu, “New challenges for biological text-mining in the next decade,” *Journal of Computer Science and Technology*, vol. 25, no. 1, pp. 169–179, 2010.

[5] M. He, Y. Wang, and W. Li, “PPI finder: a mining tool for human protein-protein interactions,” *PLoS ONE*, vol. 4, no. 2, Article ID e4554, 2009.

[6] J. Zhang, D. Shen, G. Zhou, J. Su, and C.-L. Tan, “Enhancing HMM-based biomedical named entity recognition by studying special phenomena,” *Journal of Biomedical Informatics*, vol. 37, no. 6, pp. 411–422, 2004.

[7] G. Zhou and J. Su, “Exploring deep knowledge resources in biomedical name recognition,” in *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and Its Applications*, pp. 96–99, Association for Computational Linguistics, Geneva, Switzerland, June 1996.

[8] K. Lee, Y. Hwang, and H. Rim, “Two-phase biomedical NE recognition based on SVMs,” in *Proceedings of the Workshop on Natural Language Processing in Biomedicine (ACL’03)*, pp. 33–40, Association for Computational Linguistics, Sapporo, Japan, July 2003.

[9] K.-J. Lee, Y.-S. Hwang, S. Kim, and H.-C. Rim, “Biomedical named entity recognition using two-phase model based on

SVMs," *Journal of Biomedical Informatics*, vol. 37, no. 6, pp. 436–447, 2004.

[10] J. Finkel, S. Dingare, H. Nguyen, M. Nissim, C. Manning, and G. Sinclair, "Exploiting context for biomedical entity recognition: from syntax to the web," in *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and Its Applications (JNLPBA '04)*, 2004.

[11] L. Yao, C. Sun, Y. Wu, X. Wang, and X. Wang, "Biomedical named entity recognition using generalized expectation criteria," *International Journal of Machine Learning and Cybernetics*, vol. 2, no. 4, pp. 235–243, 2011.

[12] C. Sun, Y. Guan, X. Wang, and L. Lin, "Rich features based conditional random Fields for biological named entities recognition," *Computers in Biology and Medicine*, vol. 37, no. 9, pp. 1327–1333, 2007.

[13] B. Settles, "Biomedical named entity recognition using conditional random fields and rich feature sets," in *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and Its Applications*, pp. 104–107, Association for Computational Linguistics, 2004.

[14] L. Li, R. Zhou, and D. Huang, "Two-phase biomedical named entity recognition using CRFs," *Computational Biology and Chemistry*, vol. 33, no. 4, pp. 334–338, 2009.

[15] B. Settles, "Abner: an open source tool for automatically tagging genes, proteins and other entity names in text," *Bioinformatics*, vol. 21, no. 14, pp. 3191–3192, 2005.

[16] R. McDonald and F. Pereira, "Identifying gene and protein mentions in text using conditional random fields," *BMC Bioinformatics*, vol. 6, no. 1, article S6, 2005.

[17] R. Leaman and G. Gonzalez, "BANNER: an executable survey of advances in biomedical named entity recognition," in *Proceedings of the Pacific Symposium on Biocomputing*, vol. 13, pp. 652–663, 2008.

[18] R. Stre, K. Yoshida, M. Miwa, T. Matsuzaki, Y. Kano, and J. Tsujii, "Extracting protein-interactions from text with the unified AkaneRE event extraction system," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 7, no. 3, pp. 442–453, 2010.

[19] M. Miwa, R. Saetre, Y. Miyao et al., "Extracting protein-interactions from text with the unified AkaneRE event extraction system," *International Journal of Medical Informatics*, vol. 78, pp. e39–e46, 2009.

[20] S. Kim, J. Yoon, J. Yang, and S. Park, "Walk-weighted subsequence kernels for protein-protein interaction extraction," *BMC Bioinformatics*, vol. 11, article 107, 2010.

[21] A. Airola, S. Pyysalo, J. Björne, T. Pahikkala, F. Ginter, and T. Salakoski, "All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning," *BMC Bioinformatics*, vol. 9, no. 11, article 52, 2008.

[22] Y. Niu, D. Otasek, and I. Jurisica, "Evaluation of linguistic features useful in extraction of interactions from PubMed; application to annotating known, high-throughput and predicted interactions in I^2D," *Bioinformatics (Oxford, England)*, vol. 26, no. 1, pp. 111–119, 2010.

[23] Q.-C. Bui, S. Katrenko, and P. M. A. Sloot, "A hybrid approach to extract protein-protein interactions," *Bioinformatics*, vol. 27, no. 2, pp. 259–265, 2011.

[24] K. Raja, S. Subramani, and J. Natarajan, "PPInterFinder—a mining tool for extracting causal relations on human proteins from literature," *Database*, vol. 2013, Article ID bas052, 2013.

[25] C. Baral, G. Gonzalez, A. Gitter, C. Teegarden, A. Zeigler, and G. Joshi-Topé, "Cbioc: beyond a prototype for collaborative annotation of molecular interactions from the literature," in *Proceedings of the Computational Systems Bioinformatics/Life Sciences Society Computational Systems Bioinformatics Conference*, 2007.

[26] R. Hoffmann and A. Valencia, "Implementing the iHOP concept for navigation of biomedical literature," *Bioinformatics*, vol. 21, no. 2, pp. ii252–ii258, 2005.

[27] T. Munkhdalai, M. Li, E. Namsrai, O.-E. Namsrai, and K. H. Ryu, "BFSM: finite state machine learned as name boundary definer for bio named entity recognition," in *Proceedings of the 3rd International Conference on Awareness Science and Technology (iCAST '11)*, pp. 344–349, IEEE, Dalian, China, September 2011.

[28] C. N. Arighi, B. Carterette, K. Bretonnel Cohen et al., "An overview of the BioCreative 2012 workshop track III: interactive text mining task," *Database*, vol. 2013, Article ID bas056, 2013.

[29] L. Hirschman, A. Yeh, C. Blaschke, and A. Valencia, "Overview of BioCreAtIvE: critical assessment of information extraction for biology," *BMC Bioinformatics*, vol. 6, no. 1, article S1, 2005.

[30] L. Smith, L. K. Tanabe, R. J. Ando et al., "Overview of BioCreative II gene mention recognition," *Genome Biology*, vol. 9, supplement 2, article S2, 2008.

[31] D. Tikk, P. Palaga, and U. Leser, "A fast and effective dependency graph kernel for PPI relation extraction," *BMC Bioinformatics*, vol. 11, article 8, 2010.

[32] D. Otasek, K. Brown, and I. Jurisica, "Confirming protein-protein interactions by text mining," in *Proceedings of the 6th SIAM Conference on Text Mining*, IEEE, Bethesda, Md, USA, April 2006.

[33] K. Toutanova, D. Klein, C. Manning, and Y. Singer, "Featurerich part-of-speech tagging with a cyclic dependency network," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pp. 173–180, Association for Computational Linguistics, Baltimore, Md, USA, 2003.

[34] J.-D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii, "GENIA corpus—a semantically annotated corpus for bio-textmining," *Bioinformatics*, vol. 19, no. 1, pp. i180–i182, 2003.

[35] Uniprotkb/swiss-prot database, http://www.uniprot.org/.

[36] NCBI, "Ncbi entrez gene," http://www.ncbi.nlm.nih.gov/sites.

[37] A. Bordes, S. Ertekin, J. Weston, and L. Bottou, "Fast kernel classifiers with online and active learning," *The Journal of Machine Learning Research*, vol. 6, pp. 1579–1619, 2005.

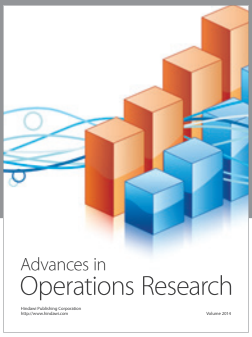
[38] Q. Bui, B. Nualláin, C. A. Boucher, and P. M. A. Sloot.

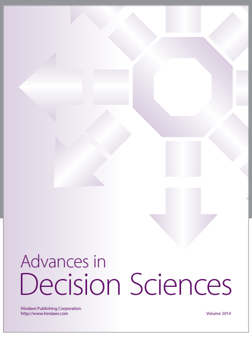
[39] F. Rinaldi, G. Schneider, K. Kaljurand, S. Clematide, T. Vachon, and M. Romacker, "Ontogene in biocreative II.5," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 7, no. 3, pp. 472–480, 2010.

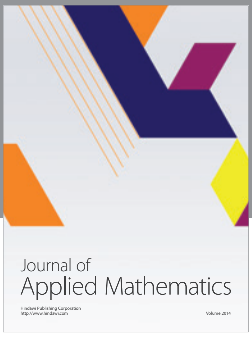
[40] K. Fundel, R. Küffner, and R. Zimmer, "RelEx—relation extraction using dependency parse trees," *Bioinformatics*, vol. 23, no. 3, pp. 365–371, 2007.

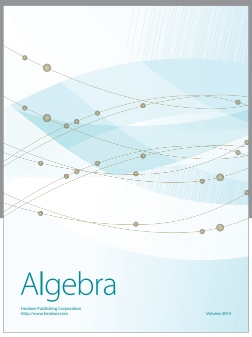
[41] J. Eom and B. Zhang, "Pubminer: machine learning-based text mining for biomedical information analysis," *Genomics & Informatics*, vol. 2, pp. 99–106, 2004.

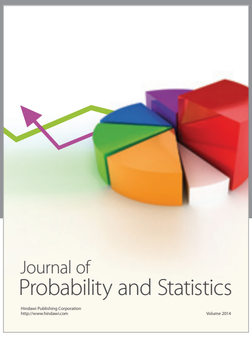
[42] S. Pyysalo, A. Airola, J. Heimonen, J. Björne, F. Ginter, and T. Salakoski, "Comparative analysis of five protein-protein interaction corpora," *BMC Bioinformatics*, vol. 9, no. 3, article S6, 2008.


Advances in
Operations Research

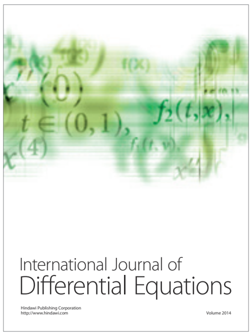

Advances in
Decision Sciences


Journal of
Applied Mathematics

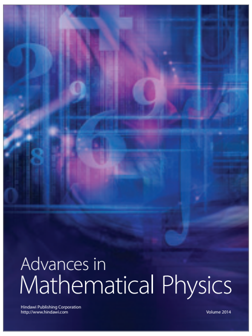

Algebra

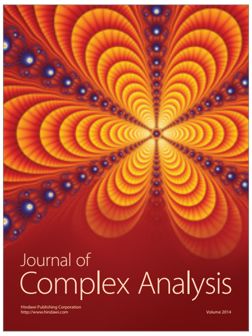

Journal of
Probability and Statistics

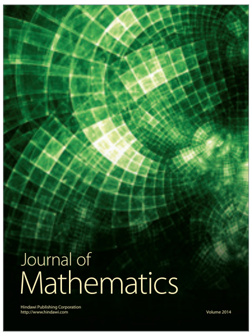
The Scientific
World Journal

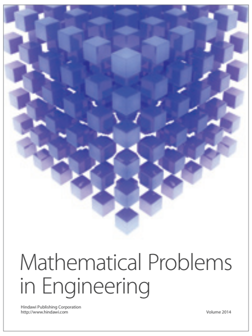

International Journal of
Differential Equations

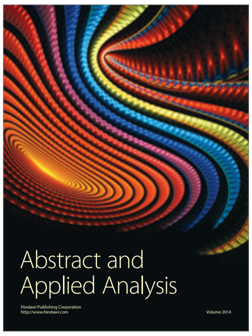
International Journal of
Combinatorics

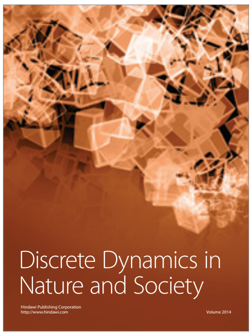

Advances in
Mathematical Physics

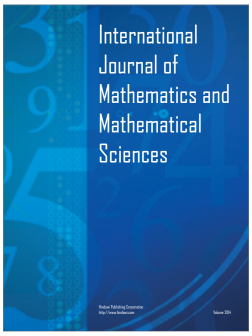

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

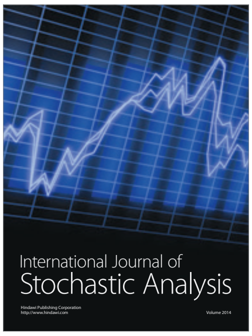

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization