



Contents lists available at ScienceDirect

Journal of Biomedical Informatics

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

Feature generation and representations for protein–protein interaction classification

Man Lan^{a,b,*}, Chew Lim Tan^c, Jian Su^b

^a East China Normal University, Shanghai, China

^b Institute for Infocomm Research, Singapore

^c School of Computing, National University of Singapore, Singapore

ARTICLE INFO

Article history:

Received 26 May 2008

Available online xxx

Keywords:

Biomedical text classification

Feature representation

Protein–protein interaction

PPI

ABSTRACT

Automatic detecting protein–protein interaction (PPI) relevant articles is a crucial step for large-scale biological database curation. The previous work adopted POS tagging, shallow parsing and sentence splitting techniques, but they achieved worse performance than the simple bag-of-words representation. In this paper, we generated and investigated multiple types of feature representations in order to further improve the performance of PPI text classification task. Besides the traditional domain-independent bag-of-words approach and the term weighting methods, we also explored other domain-dependent features, i.e. protein–protein interaction trigger keywords, protein named entities and the advanced ways of incorporating Natural Language Processing (NLP) output. The integration of these multiple features has been evaluated on the BioCreAtIvE II corpus. The experimental results showed that both the advanced way of using NLP output and the integration of bag-of-words and NLP output improved the performance of text classification. Specifically, in comparison with the best performance achieved in the BioCreAtIvE II IAS, the feature-level and classifier-level integration of multiple features improved the performance of classification 2.71% and 3.95%, respectively.

© 2009 Elsevier Inc. All rights reserved.

1. Introduction

With the rapid growth of the volume of published biological and biomedical articles, automatic detecting articles containing specific biological information relevant to the users' needs is a crucial step for large-scale biological database curation. Therefore, several biomedical and biological text classification practical activities have been presented in recent years. For example, KDD Cup 2002 [1] proposed a biological text classification task to determine whether the article contains experimental evidence of gene expression. In addition, the TREC Genomics track 2005 [2] focused on the evaluation of text classification in the context of Genomics.

In this paper, we contributed to the domain of protein–protein interaction (PPI) classification in the perspective of feature generation and integration. Protein–protein interaction information is one most important biomedical problem, which is crucial to understanding not only the functional role of individual proteins but also the organization of the entire biological processes. This work is motivated by the BioCreAtIvE II Challenge [3], an international evaluation in biological text mining, which proposed a specific Protein Interaction Article Sub-task (IAS) focusing on the detection of

protein–protein interaction relevant articles from PubMed titles and abstracts.

The representation of documents is a key aspect in all text classification approaches since effective feature representation is essential to make the learning task efficient and more accurate. Therefore, researchers have adopted several different ways to represent text for biological text classification, for example, predefined entities and keywords [4], expert-defined rules [5], local patterns [6], etc. Since these features have been examined on the gene expression information, we are interested to explore multiple features for this specific PPI information as well.

In the BioCreAtIvE II Challenge, most of the participated teams adopted traditional bag-of-words approach to represent text [7]. No advanced NLP techniques or components but stemming and stop words list were adopted in most of the teams. Even though a few of teams used POS tagging (Team 4, Team 6 and Team 41), shallow parsing (Team 4, Team 6 and Team 41) and sentence splitting (Team 6 and Team 49) [7], they have not achieved better performance than those who used the simple bag-of-words approach in terms of F_1 measure. In addition, [8] explored other complicated advanced NLP technique, such as adopting Named Entities as features, but they adopted this feature in a quite simple and straightforward way (they only check the existence of proteins in document level) and thus it has not shown a good result in comparison with bag-of-words approach. Although we (Team 57)

* Corresponding author. Address: East China Normal University Shanghai, China.

E-mail addresses: mlan@cs.ecnu.edu.cn, mlan@i2r.a-star.edu.sg (M. Lan), tancl@comp.nus.edu.sg (C.L. Tan), sujian@i2r.a-star.edu.sg (J. Su).

achieved the best performance in the IAS task of the BioCreAtIvE II Challenge in terms of F_1 measure (0.78)¹, we only adopted the simple bag-of-words approach [9]. Therefore, how to efficiently exploit more domain-dependent features in biological literature, more advanced ways of incorporating NLP output to further improve the performance motivates us to have in depth investigation in this paper.

Therefore, besides the traditional domain-independent bag-of-words approach and term weighting methods, we also explored other domain-dependent features, i.e. protein–protein interaction trigger keywords, protein named entities (NER) and advanced ways of incorporating NER output in sentence level. In addition, the integration of these multiple features from feature-level and classifier-level for this specific PPI text classification task has been evaluated on the BioCreAtIvE II corpus. To the best of our knowledge, so far no such work as incorporating NER in sentence level has been explored in the PPI task.

The rest of the paper is organized as follows. In Section 2, we present detailed descriptions of methodologies adopted in this paper. In Section 3, we report the experimental results and analysis. Finally, the conclusions are drawn in Section 4.

2. Methodology

In this paper, we focus on addressing the text classification problem by means of different ways for feature generation and integration, including domain independent knowledge (i.e. term weighting methods) and domain dependent knowledge (i.e. trigger keywords, protein named entities (PNE)). To further improve the performance by using advanced natural language processing techniques, we also stepped into the sentence level to generate more features based on protein named entities and protein interactive trigger keywords. We also explored the performance of their integration in feature level and classifier level as well. In addition, in order to check whether these different feature representations are significantly different from each other, the statistical significance tests on these feature representations and corrections for multiple comparisons have been performed.

2.1. Data corpus

The training corpus of BioCreAtIvE II challenge in year 2006 is a collection of abstracts which contains 3536 true positive documents (64.3%) relevant for PPI curation and 1959 true negative documents (35.7%) not relevant for PPI curation from two databases, i.e. IntAct and MINT. In the test period, participants received 750 unlabelled test abstracts and had to classify them and submitted the test results in one week. The data corpus of BioCreAtIvE II can be downloaded from http://biocreative.sourceforge.net/biocreative_2_dataset.html.

The Porter's stemming was performed to reduce words to their base forms. Stop words (513 stop words), punctuation and numbers were removed. The threshold of the minimal term length is 3 (since many biological keywords or acronym contain 3 letters). The resulting vocabulary has 24648 words (terms or features). By using χ^2 statistic ranking metric for feature selection, the top $P = \{200, 300, 400, 450, 500, 1000, 1500\}$ features from positive and negative categories were selected from the training data set. Since the best performance has been achieved using 900 features

(bag-of-words) in our previous experiments based on a through evaluation [9], we only reported this best result by using 900 features in this paper.

2.2. The bag-of-words approach

The most widely-used text representation for general text classification task is known as the “bag-of-words” approach. For most bag-of-words representations, each feature corresponds to a single word in the training corpus, usually with case information and punctuation removed. Often infrequent and frequent words are removed from the original text. Sometimes a list of stop words (functional or connective words that are assumed to have no information content) is also removed.

Typically, in order to make the features more statistically independent, a stemming algorithm is performed to remove suffixes from words, which has the effect of mapping several morphological forms of words to a common feature. In most cases, the stemmed root may not be a complete word.

Besides feature type, another important issue is term (i.e. feature) weighting. Different terms have different importance in a text and thus an important indicator represents how much this term contributes to the semantics of document. Term weighting methods can assign appropriate weights to terms to improve the performance of text categorization. We have earlier proposed a new effective supervised term weighting method, i.e. *tf.rf*, which has been confirmed to perform significantly better than other methods (including *tf.idf* and other supervised term weighting methods) on several widely-used newswire benchmark corpora cross different learning methods (see [10,11]). Recently, it also has been confirmed the best in other researcher's work in [12]. Therefore, in this PPI domain, we examine the results of this term weighting method as well.

2.3. Trigger keywords

Generally, trigger keywords indicate an interaction relationship between the given protein entities and trigger potential extraction patterns about PPI. The idea of using trigger keywords to extract patterns from sentences can be found in [13,14]. Typically, these trigger keywords are selected out by the biological domain experts. Table 6 in Appendix A lists 70 stemmed trigger keywords used in this paper, which are mainly from [15]. These stemmed trigger keywords are selected for several reasons. First, verb trigger keywords express existence and action of proteins and their interactions, which is based on the consideration that relevant PPI abstracts describe interaction events between proteins. Second, noun trigger keywords express the occurrence and locales of proteins and their interactions. Generally, these trigger keywords are expected to serve as a complement to feature representation and preserve more information neglected by using protein named entities feature alone. Moreover, they are expected to significantly reduce the high dimensions caused by the hundreds of bag-of-words features without decreasing the classification performance as well.

2.4. Protein named entities (PNEs)

A very basic observation about bag-of-words representation is that a great deal of information in the context from the original documents is discarded and thus the syntactic structures are also broken. The end result is that the text is represented incoherent to humans in order to make it coherent to a machine learning algorithm. On the other hand, the goal of using protein named entities (PNEs) as features is to attempt to capture some of the information left out of the bag-of-words representation, especially for this PPI classification task.

¹ This result is officially published by BioCreAtIvE II organizer after they refined the released test samples by removing 37 relevant and 36 non-relevant abstracts during the post-evaluation period. Since the published results were evaluated on a smaller and cleaner test data set (only 677 total articles), they are a bit higher than our results on the original total 750 articles in this paper. Note that BioCreAtIvE II has not published which test abstracts were removed from the initial test collection. Thus we only report the results on the full original data set.

2.4.1. Protein named entities recognition system

Recognizing named entities like gene, protein and virus, is quite important for biomedical information retrieval and information extraction. It is a challenging task because there is no standard naming conventions of named entities in the biomedical domain, being much more difficult than the one in the news domain. For example, many biomedical entity names are descriptive and have many words, numbers and special characters. In addition, one biomedical entity name may be with various spelling forms with capitalization or hyphen or even various irregular abbreviations.

In this paper, we adopted an existing named entity recognition system named PowerBioNE [16], which is based on a Hidden Markov Model (HMM). In this recognition system, various evidential features are integrated through a HMM-based recognizer to deal with various complex naming conventions in the biomedical domain. This system achieved the best performance in terms of F_1 measure (80.63%) in the protein names recognition subtask in the first BioCreAtIvE challenge [17]. Due to lack of enough annotated training corpus, we only use the PowerBioNE system trained on a general biomedical data corpus to extract protein names from the BioCreAtIvE II corpus.

2.4.2. Protein named entities distribution

The PowerBioNE recognition system has extracted 30,780 protein named entities (even more than the 24,648 words in the whole resulting vocabulary after stemming and removing stop words) from the training corpus. One noticing phenomenon of these extracted named entities is the wide distribution in the training and test data set (whether in positive or negative samples). Table 1 shows the statistics of distribution of abstracts with different number of PNEs in the training and test documents. Although the accuracy of PowerBioNE is not very high, there are some issues worthy of discussion. First, it is favorable that vast majority of PPI-relevant documents (99.1%) have at least two PNEs. Second, unfavorably, 76.68% of non-relevant articles have at least two PNEs as well. This indicates that detecting these non-relevant articles is quite challenging. That is, although these articles are not relevant to PPI information, their contents are naturally close to protein-relevant. Third, 96.53% of test instances have at least two PNEs while only half of them are PPI-relevant. This shows that these test articles are quite noisy and it is quite difficult for curators to detecting whether they are PPI-relevant.

Another noticing phenomenon is sparse occurrence. Most of the extracted protein named entities occur only once or few times in the corpus. For example, 25,740 named entities (83.7%) occur only once, 2529 entities (8.2%) occur more than three times and only 380 entities (1.2%) occur more than ten times in the whole corpus. This sparse occurrence problem makes the document indexing difficult since many documents will be represented as null vectors when the number of named entities used for indexing is quite small.

2.4.3. Simple usage of protein named entities

In our previous work in [18], we simply considered the existence of PNE in the document as one feature in a text (i.e. 0 for absence and 1 for presence) and combined PNE with the bag-of-words representation. The previous experiments showed that the

simple combination of the two feature representations has worse performance than the bag-of-words representation alone.

In consideration of the specific PPI task, a general basic idea is that since the PPI articles describe the interactive connections between proteins, there should be more (or at least two) PNEs in the relevant articles. Therefore, in this paper, we also considered the frequency of PNEs as features. Specifically, due to the wide and sparse distribution of PNEs, we first adopted the following three PNE features from abstract level for text representation to avoid the null vectors: (1) if the article has at least one PNE; (2) if the article has at least two PNEs; (3) if the article has more than two PNEs. For each of the above three features, we use 0 for NO and 1 for YES. For example, the PowerBioNE system extracted three PNEs from the article with PubMed id 1321290, i.e. *p53*, *e6* and *hvp-16*, thus its feature vector is represented as (1 1 1) in this 3-PNE representation.

2.4.4. Advanced usage of protein named entities

However, the above work which adopted the occurrence or frequency of PNEs in the abstract as features may not be quite appropriate. As shown in Table 1, most negative abstracts do contain PNEs in the documents, which makes the discriminating of positive abstracts from negative abstracts more challenging. Therefore, besides the above work which considered the frequency of PNEs in abstract level, we attempt to use PNEs from sentence level to generate more features in order to capture more information.

To further improve the accuracy of classification, we state that it is necessary to extract useful information from the sentence level in order to generate more effective PNE-relevant features. For example, in abstract level, most negative abstracts contain protein named entities and/or trigger keywords in the content even though they are not relevant to PPI information. In most cases, these PNEs and/or trigger keywords are in separate sentences. Sometimes even though two or more PNEs are in the same sentence, there is no interactive relationship indicators between them. On the contrary, the two or more PNEs in the positive abstracts would be connected by using interactive indicators. Therefore, we need to get into the sentence level and find out more useful feature representations.

To do so, we first selected out the sentences which contain at least one PNE and at least one trigger keyword from abstracts. Then by counting the frequency of interactive indicators and PNE pairwires occurring in one single sentence, we selected out 11 effective interactive indicators based on Odds Ratio metric from the 70 trigger keywords (as shown in Table 6). Table 2 lists these 11 interactive indicators selected.

Thus, we adopted the following interactive PNE features for representation: (1) if the sentence has PNE and interactive keyword; (2) if the sentence has 2 PNEs and interactive keyword; (3) if the sentence has more than 2 PNEs and interactive keyword (for each feature, 0 for NO and 1 for YES). We named this representation as *interact-PNE*.

2.5. Support vector machines

Support vector machine (SVM) is a relatively new machine learning algorithm based on the structural risk minimization prin-

Table 1

Distribution of abstracts with different numbers of protein named entities (PNEs) in the training and test corpus.

Data set	sum_docs	no_PNE	1_PNE	2_PNEs	3_PNEs
Positive training	3536	11 (0.3%)	21 (0.6%)	47 (1.3%)	3457 (97.8%)
Negative training	1959	266 (13.6%)	191 (9.8%)	206 (10.4%)	1296 (66.2%)
Test	750	13 (1.7%)	13 (1.7%)	36 (4.8%)	688 (91.8%)

Table 2

Eleven interactive indicators selected from 70 trigger keywords based on Odds Ratio metric.

associ interact	assembl interfac	bind intra	complex residu	disassembl surface	inter
--------------------	---------------------	---------------	-------------------	-----------------------	-------

principle from computational learning theory, which seeks, among all the surfaces in $|W|$ -dimensional ($|W|$ is the number of features) space that separate the training data examples into two classes, the surface (decision surfaces) that separates the positives from the negatives by the widest possible margin. Thus this best decision surface is determined by only a small set of training examples, known as support vectors. This quite interesting property makes SVM theoretically unique and different from many other methods, such as kNN, Neural Network and Naive Bayes where all the data examples in the training data set are used to optimize the decision surface [19].

In recent years, SVM has been extensively used in text classification and has been confirmed to show better performance than other conventional machine learning algorithms to handle relatively high dimensional and large-scale training set ([19–22]). Specifically, our benchmark adopted the linear SVM rather than non-linear SVM. The reasons why we chose linear kernel function of SVM in our experiments are listed as follows. First, linear SVM is simple and fast [21]. Second, linear SVM performs better than the non-linear models [19,21]. The SVM software we used is LIBSVM-2.8 [23].

2.6. Performance evaluation

Classification effectiveness is usually measured by using *precision* (P) and *recall* (R). Neither *precision* nor *recall* makes sense in isolation from each other as it is well known from the information retrieval practice that higher levels of precision may be obtained at the price of low values of recall. Thus, a classifier should thus be evaluated by means of F_1 function which attributes equal importance to *precision* and *recall*. Typically, the *precision*, *recall*, F_1 and *accuracy* have been calculated as follows:

$$\text{precision} = \frac{TP}{TP + FP} \quad (1)$$

$$\text{recall} = \frac{TP}{TP + FN} \quad (2)$$

$$F_1 = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3)$$

$$\text{accuracy} = \frac{TP + TN}{N} \quad (4)$$

where TP, number of True Positive predictions; FP, False Positives; TN, True Negatives; FN, False Negatives; N, total number of Positives and Negatives in the data set.

2.7. Statistical significance tests and multiple-comparison correction

To verify the impact of the difference on the performance variation of these different feature representations and their integrations, we employed the McNemar's significance tests [24]. McNemar's test is a χ^2 -based significance test for goodness of fit that compares the distribution of counts expected under the null

hypothesis to the observed counts. Two classifiers f_A and f_B based on two different text representations were performed on the test set. For each example in test set, we recorded how it was classified and constructed the following contingency table (Table 3). The null hypothesis for the significance test states that on the test set, two classifiers f_A and f_B will have the same error rate, which means that $n_{10} = n_{01}$. Then the statistic χ is defined as:

$$\chi = \frac{(|n_{01} - n_{10}| - 1)^2}{n_{01} + n_{10}} \quad (5)$$

where n_{10} and n_{01} are defined in Table 3. Dietterich showed that under the null hypothesis, χ is approximately distributed as χ^2 distribution with 1 degree of freedom, where the significance levels 0.01 and 0.001 corresponded to the two thresholds $\chi_0 = 6.64$ and $\chi_1 = 10.83$, respectively. Given a χ score computed based on the performance of a pair of classifiers f_A and f_B , we compared χ with threshold values χ_0 and χ_1 to determine if f_A is superior to f_B at significance levels of 0.01 and 0.001, respectively. If the null hypothesis is correct, then the probability that this quantity is greater than 6.64 is less than 0.01. Otherwise we may reject the null hypothesis in favor of the hypothesis that the two text representations have different performance.

In order to eliminate the tendency for multiple comparisons to yield spurious significant differences and make the multiple pairwise comparisons more reliable and reasonable, we also applied a multiple-comparison procedure called Holm test [25] which includes appropriate corrections for the fact that we are comparing more than one pair of means. The Holm test is a so-called sequentially rejective or step-down, procedure because it applies an accept/reject criterion to a set of ordered null hypotheses, starting with the smallest p value (probability value of the statistic test), and proceeding until it fails to reject a null hypothesis.

3. Experimental results and discussion

Table 4 lists the detailed results in terms of precision, recall, F_1 and accuracy of different feature representations, their integrations and the best three results performed by previous participants for BioCreAtIvE II task, where 900BOW means using 900 words (bag-of-words), 3PNE means using 3 PNEs, 70trigger means using 70 trigger keywords, interact-PNE means integration of PNE and interactive indicators in single sentence level. Their combinations are denoted by using "+" sign. For most of each representation, we also tried two different term weighting methods, i.e. the binary and the *tf.rf* method.

In many cases, a single feature is easy to lead the classifier's over-dependency on the data, thus different features may complement each other. That is, the false judgments caused by one feature would be treated correctly by another one. Therefore, we performed feature integration in two different levels, i.e. feature level and classifier level. Specifically, Run 7–11 are feature-level integration, i.e. different features are normalized respectively first before they are combined together into a new feature vector. Run 12–14 are classifier-level integration, which is also known as *majority voting* scheme.

Note that Run 2, i.e. the system based on 900 words weighted by *tf.rf* is actually the system configuration that we (Team 57) achieved the best F_1 performance in the BioCreAtIvE II Challenge. Here it serves as the baseline for comparison. Moreover, to make

Table 3

McNemar's test contingency table.

n_{00} : Number of examples misclassified by both classifiers f_A and f_B
 n_{10} : Number of examples misclassified by f_B but not by f_A

n_{01} : Number of examples misclassified by f_A but not by f_B
 n_{11} : Number of examples misclassified by neither f_A nor f_B

Table 4

Results of different feature representations, their integrations and the best three results performed by previous participants for BioCreAtIvE II task.

Run	Representation	Weighting	Precision	Recall	F ₁	Accuracy
1	900BOW	(binary)	67.32	81.87	73.89	71.07
2	900BOW	(tf.rf)	69.59	86.67	77.20	74.40
3	3PNE	(binary)	53.49	98.13	69.24	56.40
4	70Trigger	(binary)	67.41	80.53	73.39	70.80
5	70Trigger	(tf.rf)	66.81	83.73	74.32	71.07
6	interact-PNE	(binary)	66.21	90.40	76.44	72.13
7	70Trigger + 3PNE	(binary)	67.76	82.40	74.37	71.60
8	70Trigger + 3PNE	(tf.rf)	67.79	85.87	75.76	72.53
9	900BOW + 70Trigger + 3PNE	(binary)	67.69	82.13	74.22	71.47
10	900BOW + 70Trigger + 3PNE	(tf.rf)	69.21	86.93	77.07	74.13
11	900BOW + interact-PNE	(tf.rf)	70.59	90.43	79.29	76.60
<i>Classifier integration</i>						
12	Run: 3 + 6 + 8		58.48	86.40	69.75	62.53
13	Run: 2 + 3 + 5		65.28	92.27	76.46	71.60
14	Run: 2 + 3 + 10		71.81	90.93	80.25	77.40
<i>Top</i>						
<i>Team No. + Run No.</i>						
1	T57-run1		70.31	87.57	78.00	75.33
2	T28-run1		75.07	81.07	77.95	77.10
3	T57-run2		70.24	87.28	77.84	75.18

the comparison meaningful, we also listed the reported top three results in terms of F_1 value from the BioCreAtIvE II Challenge workshop [7], of which we (Team 57) achieved the first and third place and Team 28 achieved the second place. Although the system configuration of Run 2 is actually the same as Top 1, the reported result of Top 1 is a bit better than Run 2. This is due to the variance of test samples. That is, the Top 3 results were achieved after the BioCreAtIvE II organizer refined the released test corpus by removing 37 relevant and 36 non-relevant abstracts during the post-evaluation period. Since the published released results were evaluated on a smaller and cleaner data set (only 677 total articles), they are supposed to be a bit better than our results on the total 750 articles in this paper. Note that BioCreAtIvE II has not published which test abstracts were removed from the initial test collection. Thus we only report the results on the full original data set.

Regarding to the 14 feature representations, we first adopted the McNemar's tests [24] to calculate the χ statistic between any pair of two runs and to validate if there is a significant difference between two runs. In order to make corrections for multiple comparisons, we then performed the Holm test by making 91 ($14 \times 13 / 2 = 91$) pairwise comparisons (as there are 14 runs in experiment). Table 5 summarizes the χ statistic values for pairwise comparison on 14 runs using the McNemar's significance tests. Obviously, this

matrix is a symmetric triangular matrix and the values of all diagonal elements are 0.

Given the χ statistic values of pairwise comparison and the subsequent corrections for multiple comparisons, we list an approximate rank of these runs as follows:

$$\{11, 14\} > \{2, 8, 10\} > \{1, 4, 5, 6, 7, 9, 13\} \gg \{12\} \gg \{3\}$$

The runs with insignificant performance differences are grouped into one set and ">" and ">>" denote better than at significance levels of 0.01 and 0.001, respectively. Although we group these runs into different sets, the borders between some of them are fuzzy, for example, the χ statistic value between Run 1 and Run 8 is 5.26, which is smaller than the threshold value $\chi_0 = 6.64$ and thus we cannot determine that Run 1 is superior to Run 8 at the significance level of 0.01. However, since most of them have shown consistent performance with respect to each other, this approximate rank is reasonable and some interesting observations can be found as follows.

First, from feature level, Run 11, i.e. the integration of bag-of-words approach (weighted by *tf.rf*) and the interactive-PNE representation achieved the best performance in terms of precision, F_1 and accuracy, and rather good performance in terms of recall among these different features. This result is also superior to the best results in the BioCreAtIvE II challenge which we achieved

Table 5The χ statistic values for pairwise comparison on 14 runs using the McNemar's significance tests. Each cell of this pairwise comparison matrix, i.e. C_{ij} , represents the χ statistic value of Run i and Run j .

Run	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0.00	7.00	51.21	0.13	0.57	4.88	0.25	5.26	0.00	8.52	11.80	40.50	0.12	13.52
2	—	0.00	81.25	12.00	6.16	5.69	4.76	3.50	5.50	0.25	7.56	83.10	6.35	7.78
3	—	—	0.00	47.70	54.50	77.78	55.04	67.61	53.84	80.29	107.66	15.11	80.82	116.27
4	—	—	—	0.00	0.05	1.27	3.13	5.33	2.29	19.36	20.93	35.10	0.30	24.58
5	—	—	—	—	0.00	4.17	0.64	3.45	0.27	8.00	11.56	47.25	4.15	10.52
6	—	—	—	—	—	0.00	1.16	1.11	1.28	5.78	31.03	23.35	0.50	30.63
7	—	—	—	—	—	—	0.00	1.89	0.00	10.32	13.08	45.81	0.01	18.20
8	—	—	—	—	—	—	—	0.00	2.45	3.75	7.30	69.32	0.65	9.00
9	—	—	—	—	—	—	—	—	0.00	11.25	13.92	44.00	0.00	18.89
10	—	—	—	—	—	—	—	—	—	0.00	6.72	77.29	5.49	7.64
11	—	—	—	—	—	—	—	—	—	—	0.00	52.15	25.41	0.57
12	—	—	—	—	—	—	—	—	—	—	—	0.00	17.78	52.94
13	—	—	—	—	—	—	—	—	—	—	—	—	0.00	34.24
14	—	—	—	—	—	—	—	—	—	—	—	—	—	0.00

Table 6
70 stemmed trigger keywords

accumul	activ	add	addit	addition	apoptosi	associ
bind	block	bound	catalyz	cleav	complex	contain
decreas	demethyl	dephosphoryl	deplet	disassembl	discharg	domain
downregul	down-regul	elev	express	impair	inact	inactiv
increas	induc	induct	influen	inhibit	initi	inter
interact	interfac	intra	involv	mediat	methylyat	modif
modifi	modul	myogenesi	overexpress	particip	phosphoryl	produc
product	phosphorylat	promot	protein	react	reduc	reduct
regul	regulat	releas	replac	repress	residu	secret
sever	stimul	substitut	surfac	transactiv	upregul	up-regul

[9] (Run 2) and improved the F_1 measure 2.71%. Moreover, each of these two feature representations themselves has better performance than other single feature representation as well. The consequent statistical significance tests also confirmed that these two methods and their integration are superior to other single feature representations. In addition, the statistical significance tests indicated that the *tf.rf* method is superior to the binary method at significance level of 0.01. This result once again confirmed the effectiveness of *tf.rf* weighting method for text classification. Furthermore, compared with the 3-PNE representation, the good performance of interactive-PNE comes from features generated from the sentence level. This observation indicated that the sentence-level information is essential to the accuracy of classification performance and complement the abstract-level features as well.

Second, using 3-PNE representation alone achieves the worst F_1 value among all the feature representations. However, the 3-PNE representation has the highest recall value among all the feature representations, i.e. 98.13%. In some real-life scenarios where the underlying end user demands do not focus on the F_1 value only, for example in case of exhaustive curation, a high recall might be more desirable. Thus, the 3-PNE representation would be favorable since it only uses 4 features to represent all the articles and consequently it is quite efficient for the on-line curation system.

Third, interestingly, the trigger keywords representation method has much less features (70 words) than the bag-of-words approach (900 words) but it achieved acceptable performance. This observation is interesting since in real-life application, the detecting system will benefit from less features and faster indexing and predicting process. However, when combined with 3-PNE representation, the 70 trigger keywords representation has not made any significant improvement. This result is beyond our original expectation that this combination of trigger keywords and PNE would capture more information than the bag-of-words approach or 70 trigger keywords or 3-PNE alone.

Finally, we also adopted a simple majority voting technique in order to further improve the performance. The classifier-level feature integration results also showed that Run 14, i.e. the integration of bag-of-words approach (weighted by *tf.rf*), the interactive-PNE representation and the 3-PNE representation, achieved the best performance in terms of precision, F_1 and accuracy. In comparison with the best performance achieved in the BioCreAtIvE II IAS, this classifier-level integration of multiple features improved the performance of classification 3.95%. It indicated that the *tf.rf* weighting method and sentence-level feature generation also complement each other in the classifier-level as well.

4. Conclusions

In this paper, we examined multiple feature representations for protein–protein interaction classification task, i.e. bag-of-words approach with different term weighting schemes, protein named entities, trigger keywords, sentence level feature generation, and their combination. The experimental results are encouraging. The

integration of the bag-of-words approach (weighted by *tf.rf*) and the features generated from sentence-level using advanced NLP techniques (protein named entities) achieved the best performance from two ways of feature integration. In comparison with the best performance achieved in the BioCreAtIvE II IAS, the feature-level and classifier-level integration of multiple features improved the performance of classification 2.71% and 3.95%, respectively.

We should point out that the observations above are made based on the controlled experiments and the accuracy of extracted protein named entities also has an effect on the result. This work encouraged our future work on more advanced NLP techniques and advanced ways of incorporating NLP output to further improve the performance of text classification, for example, high performance coreference resolution to normalize the protein names through different variations, nominal or pronominal expressions could generate more occurrences of the same protein names to facilitate the further text classification.

Acknowledgment

The work was supported by a Shanghai Pujiang Talent Program 09PJ1404500.

Appendix A. List of 70 stemmed trigger keywords

See Table 6.

References

- [1] Yeh A, Hirschman L, Morgan A. Background and overview for KDD Cup 2002 task 1: information extraction from biomedical articles. SIGKDD Explor Newsl 2002;4(2):87–9.
- [2] Hersh W, Cohen A, Yang J, Bhupatiraju RT, Roberts P, Hearst M. Genomics track overview. In: Proceedings of the 14th Text REtrieval Conference (TREC 2005), 2005.
- [3] BioCreAtIvE II 2006. Available from: <http://biocreative.sourceforge.net/>.
- [4] Keerthi SS, Ong CJ, Siah KB, Lim DB, Chu W, Shi M, et al. A machine learning approach for the curation of biomedical literature: KDD Cup 2002 (task 1). SIGKDD Explor Newsl 2002;4(2):93–4.
- [5] Regev Y, Finkelstein-Landau M, Feldman R, Gorodetsky M, Zheng X, Levy S, et al. Rule-based extraction of experimental evidence in the biomedical domain: the KDD Cup 2002 (task 1). SIGKDD Explor Newsl 2002;4(2):90–2.
- [6] Ghanem MM, Guo Y, Lodhi H, Zhang Y. Automatic scientific text classification using local patterns: KDD CUP 2002 (task 1). SIGKDD Explor Newsl 2002;4(2):95–6.
- [7] Krallinger M, Valencia A. Evaluating the detection and ranking of protein interaction relevant articles: the BioCreative challenge interaction article sub-task (IAS). In: Proceedings of the second BioCreAtIvE challenge workshop 2007; p. 29–39.
- [8] Grover C et al. Adapting a relation extraction pipeline for the BioCreAtIvE II tasks. In: Proceedings of the second BioCreative challenge evaluation workshop 2007; p. 273–86.
- [9] Man Lan, Chew Lim Tan, Jian Su. A term investigation and majority voting for protein interaction article sub-task 1 (IAS). In: Proceedings of the second BioCreative challenge evaluation workshop 2007; p. 183–5.
- [10] Man Lan, Chew Lim Tan, Hwee Boon Low. Proposing a new term weighting scheme for text categorization. In: Proceedings of the 21st national conference on artificial intelligence (AAAI'06) 2006.

- [11] Man Lan, Chew Lim Tan, Jian Su, Yue Lu. Supervised and traditional term weighting methods for automatic text categorization. *IEEE transactions on pattern analysis and machine intelligence* 2009;31(4):721–35.
- [12] Richard Tzong-Han Tsai, Hsieh-Chuan Hung, Hong-Jie Dai, Yi-Wen Lin. Protein–protein interaction abstract identification with contextual bag of words. In the 2nd International symposium on languages in biology and medicine, Singapore 2007.
- [13] Agichtein E, Gravano L. Snowball: extracting relations from large plain-text collections. In: *Proceedings of the fifth ACM international conference on digital libraries* 2005; p. 85–94.
- [14] Krallinger M, Padron M, Blaschke C, Valencia A. Assessing the correlation between contextual patterns and biological entity tagging. In: *Proceedings of COLING-NLPBA 2004*; p. 36–43.
- [15] Temkin JM, Gilder RM. Extraction of protein interaction information from unstructured text using a context-free grammar. *Bioinformatics* 2003;19(16):2046–53.
- [16] GuoDong Zhou, Jian Su. Exploring deep knowledge resources in biomedical name recognition. In: *Proceedings of JNLPBA shared task 2004*; p. 99–102.
- [17] Zhou GuoDong, Shen Dan, Zhang Jie, Su Jian, Tan Soon Heng and Tan Chew Lim. Recognition of protein/gene names from text using an ensemble of classifiers and effective abbreviation resolution. In: *Proceedings of the first BioCreative challenge evaluation workshop 2004b*.
- [18] Man Lan, Chew Lim Tan. The integration of multiple feature representations for protein–protein interaction classification task. In: *Second International Symposium on Languages in Biology and Medicine, Singapore 2007*.
- [19] Yang Y, Liu X. A re-examination of text categorization methods. In: *Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval*, ACM Press 1999; p. 42–9.
- [20] Thorsten Joachims. Text categorization with support vector machines: learning with many relevant features. In: *Proceedings of ECML-98, Chemnitz, DE 1998*, p. 137–42.
- [21] Dumais S, Platt J, Heckerman D, Sahami M. Inductive learning algorithms and representations for text categorization. In: *Proceedings of the seventh international conference on Information and knowledge management*, ACM Press 1998; p. 148–55.
- [22] Leopold Edda, Kindermann Jorg. Text categorization with support vector machines: how to represent texts in input space? *Machine Learning* 2002;46(1–3):423–44.
- [23] Chang C-C, Lin C-J. LIBSVM: a library for support vector machines. Software available from: <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [24] Dietterich Thomas G. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Comput* 1998;10(7):1895–923.
- [25] Holm S. A simple sequentially rejective multiple test procedure. *Scand J Stat* 1979;6:65–70.