Protein-Protein Interaction Abstract Identification with Contextual Bag of Words

Abstract

Background

This paper is concerned with the identification of biomedical abstracts related to protein-protein interactions. We propose a novel feature representation scheme, contextual-bag-of-words, to exploit protein name information.

Results

Our method outperforms well-known methods that use protein name information as additional features. We further improve performance by extracting reliable and informative instances from unlabeled and likely positive data to provide additional training data. We employ F-measure and the area under a receiver operating characteristic curve (AUC) to measure the classification and ranking abilities, respectively. Our final model achieves an F-measure of 80.34% and an AUC score of 88.06%, which are higher than those of the top-ranking system in BioCreAtIvE-II by 2.34% and 2.52%, respectively.

Conclusions

These results show the effectiveness of our contextual-bag-of-words scheme and suggest that our system could serve as an efficient preprocessing tool for modern PPI database curation.

Background

Most biological processes, including metabolism and signal transduction, involve large numbers of proteins and are usually regulated through protein-protein interactions (PPI). It is therefore important to understand not only the functional roles of the individual proteins involved but also the overall organization of each biological process [1].

Several experimental methods can be employed to determine whether a protein interacts with another protein. Experimental results are published and then stored in protein-protein interaction databases such as BIND [2] and DIP [3]. These PPI databases are now essential for biologists to design their experiments or verify their results since they provide a global and systematic view of the large and complex interaction networks in various organisms.

Initially, the results were mainly verified and added to the databases manually. Since 1990, the development of large-scale and high-throughput experimental technologies such as immunoprecipitation and the yeast two-hybrid model has boosted the output of new experimental PPI data exponentially [4]. It becomes impossible to perform the relying curation task on the formidable number of existing and emerging publications if it relies solely on human effort. Therefore, information retrieval and extraction tools are being developed to help curators. These tools should be able to examine enormous volumes of unstructured texts to extract potential PPI information. They usually adopt a general approach: finding articles relevant to PPI first, and then extracting the relevant information from them. In this paper, we focus on the first step.

Most methods in this approach formulate the article-finding step as a text classification (TC) task, in which articles relevant to PPI are denoted as positive instances while irrelevant ones are denoted as negative. We refer to this task as the PPI-TC task from now on. One advantage of this formulation is that the machine learning (ML) methods commonly used in general TC systems such as Support vector machines [5] or Bayesian approaches [6] can be modified and applied to the problem of identifying PPI-relevant articles. In spite of this advantage, there are still two main

differences between PPI-TC and TC that might be the key challenges for further improving the performance of PPI-TC systems. We discuss them in the following two paragraphs.

Words may own different meanings according to contexts

In TC, documents are usually represented by represented by a "bag of words" (BoW). However, in PPI-TC, some words are informative only in certain contexts. For example, "bind" is more informative in indicating if an abstract is PPI-relevant when it appears in a sentence that has at least two protein names. Thus, including such contextual information in the feature representation of PPI-TC is very important.

The existence of likely data

Unlike in general TC, where documents are either categorized as relevant or irrelevant to some topic, the situation is more complicated in PPI-TC. The definition of "PPI-relevant" varies with the database for which we curate. Most PPI databases define their standard according to Gene Ontology, a taxonomy that classifies all kinds of protein-protein interactions. Each PPI database may only annotate a subset of PPI types; therefore, only some of these types will overlap with a different PPI database. In PPI databases, each existing PPI record is associated with its literature source (PMID). Figure 1 shows a PPI record of the MINT database. It shows that the article with PubMed ID:11238927 contains information about the interaction between P19525 and O75569, where P19525 and O75569 are the primary accession numbers of two proteins in the UniProt database. These articles can be treated as PPI-relevant and as true positive data. However, to employ mainstream machine-learning algorithms and improve their efficacy in PPI-TC, there are still two major challenges. The first is how to exploit the articles recorded in other PPI databases. Since other

databases may partially annotate the same PPI types as the target database, articles recorded in them can be treated as *likely positive* (LP) *data*. If more effective training data are included, feature weights will be calculated more accurately and the number of unseen features will be reduced. Considering these articles may increase the generality of the original model. The second challenge is a consequence of the first: To use likely positive data we must collect corresponding *likely negative* (LN) *data*, or the ratio of positive to negative data will become unbalanced. In the following sections, we will describe how we tackle these two challenges and discuss why our methods are effective for PPI-TC.

Synopsis

To increase the readability of this paper and introduce the terminologies that will be used in the Results, Discussions, and Conclusions sections, we here summarize the major methods, datasets, and evaluation metrics used in our experiments.

Formulation and term weighting schemes

In this paper, PPI-TC is formulated as a classification problem. Each document is transformed to a feature vector and then classified as either PPI-relevant or -irrelevant. We adopt the support vector machines (SVM) as our classification model because its efficacy has been demonstrated for binary classification tasks and allows non-binary value in feature vectors.

Following the classical BoW feature representation, a document d is represented as a term vector $\mathbf{v}$, in which each dimension v_i corresponds to a term t_i. v_i is calculated by a term-weighting function, which is very important for SVM-based TC because SVM

models are sensitive to the data scale, i.e. they are dominated by some dimensions with very wide ranges.

In addition to the simplest binary features, which only indicate the existence of a word in a document, there are currently numerous term-weighting schemes that utilize term frequency (TF), inverse document frequency (IDF) or statistical metrics information. Lan et al. [7] pointed out that the popularly-used term frequency-inverse document frequency (TFIDF) method has not performed uniformly well with respect to different data corpora. The traditional IDF factor and its variants were introduced to improve the discriminating power of terms in the traditional information-retrieval field. However, in TC, this may not be the case since the IDF factor neglects the category information of the training set. Hence, they proposed two new supervised weighting schemes, relative frequency (RF) and term frequency-relative frequency (TFRF), to improve the term's discriminating power. In these functions, each term is assigned more appropriate weights in terms of different categories.

In Table 1, we list the symbols representing the number of positive and negative documents that contain and do not contain term t_i. With this table, the schemes stated above can be defined as follows:

$$\text{Binary}(t_i, d) = \begin{cases} 1, & \text{if } t_i \in d \\ 0, & \text{otherwise} \end{cases},$$

$$\text{TF}_d(t_i) = \frac{t_i\text{'s term frequency in } d}{|d|}$$

$$\text{TFIDF}(t_i, d) = \text{TF}_d(t_i) \cdot \log \frac{w + x + y + z}{w + y}, \text{ and}$$

$$\mathrm{TFRF}(t_i, d) = \mathrm{TF}_d(t_i) \cdot \log\left(2 + \frac{w}{y}\right)$$

Methods of exploiting contextual information

A PPI abstract must contain some protein names. Hence, recognition of protein names in abstracts can improve the identification of PPI abstracts. In the following paragraphs, we describe the three methods that extend the classical BoW scheme, including our proposed CBoW, along with the other two well-known methods, BoP and BoN.

Contextual bag of words (CBoW)

The number of protein names that exists in the context affects a word's informativeness for PPI relevance. Based on this fact, we distinguish the original word bags into different contextual bags. The words in individual sentences are bagged according to the number of protein names (PNs) in the sentence. If there are 0 PN, the words are put into contextual Bag 0; if 1 PN, then Bag 1; and if 2 or more PNs, then Bag 2.

Bag of phrases (BoP)

[8] suggested that adding phrases into the original bag can help retain some order information which is lost in BoW. In our case, we add PN phrases into the bag.

Bag of normalized PNs (BoN)

The more protein names that appear in an abstract, the more likely it is to be PPI-relevant. Following [9], we replace each PN in a given abstract with "PROTEIN_*i*", where *i* denotes the order of appearance in this abstract. Abstracts containing different numbers of PNs have different normalized PN features.

Utilizing the likely data

The key steps of utilizing the likely data include selecting the most effective ones and exploiting them for improving the PPI-TC model. For the first step, the LP data can be collected from other PPI databases while the LN data are not available. Therefore, collecting LP data is much easier than LN data. In our method, we choose MEDLINE abstracts in Genomic TREC 2004 collection that are not recorded in major PPI databases to be the LN data. This is because we observe that most Medline abstracts are not relevant to PPI. Then, the method described in the "**Selecting the most effective likely positive and negative data**" subsection is employed to pick the most effective likely data. The selected LP and LN data are denoted as LP* and LN* from now on. For the second step, we employ the hierarchical model that is detailed in the "**Exploiting the selected likely positive and negative data**" subsection**.**

Datasets

In our experiment, we use the dataset of the BioCreAtIvE II IAS subtask [1] because the training set contains not only the true positive data (TP) and true negative data (TN) but also the likely positive data (LP), which is very necessary for our PPI-TC system. The TP (PPI-relevant) data were derived from the content of the IntAct [10] and MINT [11] databases, which are not organism specific. TN data were also provided by MINT and IntAct database curators. The LP data comprise a collection of PubMed identifiers of articles that have been used to annotate protein interactions by other interaction databases (namely BIND [2], HPRD [12], MPACT [13] and GRID [14]). Note that this additional collection is a noisy dataset and thus not part of the ordinary TP collection, as these additional databases may have different annotation standards from MINT and IntAct (e.g. regarding the curation of genetic interactions). We randomly selected 105,000 abstracts from the Genomic TREC 2004 collection be

the LN data. It consisted of 10-year (from 1994 to 2003) published MEDLINE abstracts (4,591,008 records). The test set is a balanced dataset, which contains 338 and 339 abstracts for TP and TN respectively. According to BioCreAtIvE-II's official statement, the keyword set of the test set differs from that of the training set in order to prevent over-fitting systems from achieving unfairly high scores. The size of each dataset is shown in Table 2.

Evaluation metrics

We employ the official evaluation metrics of BioCreAtIvE II, which assess not only the accuracy of classification but also the quality of ranking of relevant abstracts.

Classification metrics

The classification metrics examine the prediction outcome from the perspective of binary classification. The value terms used in the following formulas are defined as follows: True Positive (TP) represents the number of correctly classified relevant instances, False Positive (FP) the number of incorrectly classified irrelevant instances, True Negative (TN) the number of correctly classified irrelevant instances, and finally, False Negative (FN) the number of incorrectly classified relevant instances.

The classification metrics used in our experiments are precision, recall and F-measure. The F-measure is a harmonic average of precision and recall. These three metrics are defined as follows:

$$\text{Precision} = \frac{TP}{TP + FP}, \ \text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F} - \text{measure} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Ranking metrics

Curation of PPI databases requires a classifier to output a ranked list (as opposed to a binary decision) of all testing instances based on the likelihood that they will be in the positive class. The curators can then either specify a cutoff to filter out some articles on the basis of their experience, or give higher priority to more highly ranked instances.

The ranking metric used in our experiments is AUC, the area under the receiver operating characteristic curve (ROC curve). The ROC curve is a graph of the fraction of true positives (TPR, true positive rate) vs. the fraction of false positives (FPR, false positive rate) for a classification system given various cutoffs for output likelihoods, where

$$TPR = \frac{TP}{TP + FP},\ FPR = \frac{FP}{TP + FP}$$

When the cutoff is lowered, more instances are considered positive. Hence, both TPR and FPR are increased since their numerators become larger but their denominator, denoting the total number of positive instances, remains constant. The more positive instances that are ranked above the negative ones by the classification system, the faster that TPR grows in relation to FPR as the cutoff descends. Consequently, higher AUC values indicate more reliable ranking results.

Results

Evaluation protein name information

Table 3 shows the results of different methods of exploiting PN information. CBoW significantly outperform BoW in terms of F-measure and AUC, whereas the other two configurations that incorporate PN features into BoW only slightly improve the

performance of BoW regardless of the weighting schemes. These results suggest that our idea of dividing the word bag according to a word's context is effective. Notably, the RF weighting function consistently outperforms the other two in all methods. These results demonstrate RF's appropriateness for both TC and PPI-TC.

Expanding the training set

In this section, we examine the effects of adding LP* and LN*. Using the procedure described in Methods (note: it is in the last section of this paper), we select 8,862 abstracts from the original LP dataset and 10,000 abstracts from the unlabeled data set to form the LP* and LN* datasets, respectively.

Without lost of generality, we use the CBoW feature representation scheme. Table 4 shows that irrespective of the weighting scheme used, adding the selected data improves both the F-measure and AUC. These results suggest that exploiting LP and unlabeled data not only refines the filtering accuracy but also the ranking quality effectively, which is critical for PPI database curation. Similar to the results shown in Table 3, RF also outperforms the other weighting schemes.

Compared with BioCreAtIvE-II systems

Table 5 compares our scores with the best and median scores in BioCreAtIvE-II. We can see that our system performs better than BioCreAtIvE-II's best system and significantly better than BioCreAtIvE-II median system. These results suggest that our system has state-of-the-art ability to filter out PPI-irrelevant abstracts and rank PPI-relevant ones.

Discussion

In this section, we explain CBoW's effectiveness by illustrating and analyzing feature weights in different contextual bags. First, we list the words with the largest discriminative power difference enhanced by CBoW. In an SVM model, a feature's discriminative power correlates positively to its weight. Therefore, we list the words with the largest weight variances among all bags, as shown in Table 6. We can see that these words are really the words highly related to PPI when they appear in sentences with more than two PNs.

To further explain how CBoW correctly identifies a PPI-relevant abstract, we exhibit two examples in Table 7. The words in Table 6 are marked in italic. In addition, protein names are underlined to indicate context types.

The first example (PMID=9707401) is mislabeled by BoW since it has a PPI keyword, *interaction*. However, in CBoW, only the occurrences located in the sentence with two or more protein names have high weight to indicate an abstract's PPI-relevance. The first example is not this case. Therefore, it is correctly classified by CBoW as PPI-irrelevant.

The second example (PMID=16286467) is misclassified as PPI-irrelevant by BoW because it does not contain top discriminative words such as *interaction*. However, in CBoW, the weights of *stimulation*, *regulated*, and *phosphorylation* are significantly enlarged. Therefore, it can be correctly identified as PPI-relevant.

After examining the weights of individual words in different bags, we compare the mean and standard deviation of weights for different bags (Table 8). We can see that

Bag 2 has the largest mean weight. This result is in accordance with our intuition that words in Bag 2 have the strongest discriminative power.

We then use Mann-Whitney's rank sum test and F-test to test the equality of means and variances of weights between any two bags. The *p*-values of all the tests are listed in Table 9. An extremely small *p*-value (<0.01) is considered strong support for the significant difference between the two compared distributions. According to the test results, we can see that the weights in Bag 2 and Bag 1 are significantly greater than those in Bag 0. Also, the variance of weights in Bag 2 is significantly greater than in Bag 1 and Bag 0, suggesting that the weights in Bag 2 range more widely, thus making the features in Bag 2 more discriminative and dominant.

Conclusions

In this paper, we propose a novel CBoW feature representation scheme and demonstrate its effectiveness over other methods that also exploit PN information in PPI-TC. We also develop a method to extract likely positive and likely negative data which is applicable to PPI-TC. Recently, many advanced document representation schemes have been developed. Most of them were produced by incorporating NLP-based features. [15] pointed out that these features can help disambiguate words in the bag but did not find features that are generally effective. The results of our experiments on BoP and BoN support this claim. In our method, we need to split the feature space according to different types of contexts defined by domain knowledge. Our study of the PPI-TC problem presents a potential new way of exploiting NLP-based contextual information. In the future, we will examine the generality of this idea by applying it to TC in other domains.

In targeting to an annotation standard of a specific PPI database, all other related resources can be regarded as likely-positive. In this case, the complicated dataset integration problem can be converted into an easy filtration. Also, we can extract abundant likely-negative instances from unlimited unlabeled data to balance the training data.

With our methods, our PPI-TC system has higher F-score and AUC than the rank 1 system of these metrics in the BioCreAtIvE-II IAS challenge, which suggests that our system can serve as an efficient preprocessing tool for curating modern PPI databases.

Methods

In this section, we first introduce the machine-learning model used in our system: support vector machines. Secondly, we describe how our system filters out ineffective likely-positive data and selects effective likely-negative data from unlabeled data. Finally, we explain how we exploit the selected likely-positive and negative data.

Support vector machines

The support vector machine (SVM) model is one of the best known ML models that can handle sparse high dimension data, which has been proved useful for text classification [16]. It tries to find a maximal-margin separating hyperplane $<\mathbf{w}, \varphi(\mathbf{x})> - b = 0$ to separate the training instances, i.e.,

$$\min \| \mathbf{w} \|^2 + C\sum_i \xi^{(i)} \text{ subject to}$$

$$y^{(i)}(<\mathbf{w}, \varphi(\mathbf{x}^{(i)})> - b) \geq 1 - \xi^{(i)}, \ \forall i$$

where $\mathbf{x}^{(i)}$ is the ith training instance which is mapped into a high-dimension space by $\varphi(\cdot)$, $y_i \in \{1, -1\}$ is its label, $\xi^{(i)}$ denotes its training error, and C is the cost factor

(penalty of the misclassified data). The mapping function $\varphi(\cdot)$ and the cost factor C are the main parameters of a SVM model.

When classifying an instance **x**, the decision function f(**x**) indicates that **x** is "above" or "below" the hyperplane. [17] shows that the f(**x**) can be converted into an equivalent dual form which can be more easily computed:

$$\text{primal form: } f(\mathbf{x}) = \text{sign}(\langle \mathbf{w}, \varphi(\mathbf{x}) \rangle - b)$$

$$\text{dual form: } f(\mathbf{x}) = \text{sign}\left(\sum_i \alpha^{(i)} y^{(i)} K(\mathbf{x}^{(i)}, \mathbf{x}) - b\right)$$

where $K(\mathbf{x}^{(i)}, \mathbf{x}) = \langle \varphi(\mathbf{x}^{(i)}), \varphi(\mathbf{x}) \rangle$ is the kernel function and $\alpha^{(i)}$ can be thought of as w's transformation.

In our experiment, we choose the following linear kernel because the literature had shown that this kernel is efficient and effective for TC:

$$K(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \langle \mathbf{x}^{(i)}, \mathbf{x}^{(j)} \rangle$$

which is equivalent to

$$\varphi(\mathbf{x}^{(i)}) = \mathbf{x}^{(i)}$$

Finally, the cost factor C is chosen to be 1, which is fairly suitable for most problems.

Selecting the most effective likely positive and negative data

The limited training set contains only limited numbers of true-positive (TP) and true-negative (TN) data. To increase the generality of the classification model, more external resources should be introduced. One important resource is another PPI database; abundant PPI articles are recorded in various such databases. However, most of them only annotate a selection of all the PPI types defined in Gene Ontology. Therefore, some annotations may match the criteria of the target PPI database while

others may not. This means that abstracts annotated in that database can only be treated as likely-positive examples, some of which may need to be filtered out.

Another problem is that there are no negative data or even likely-negative data in any curation. We will obtain a model with a bias toward positive prediction if only those instances in the PPI databases are used because most machine-learning-based classifiers tend explicitly or implicitly to record the prior distribution of positive/negative labels in the training data. As explained in the introduction, an imbalance in training data can cause serious problems. However, a large proportion of the biomedical literature is negative, which is exactly the opposite. Therefore, more likely-negative (LN) instances should be incorporated to balance the training data, and this can be carried out in a manner similar to filtering out LP instances.

Liu et al. [18] provide a survey of these bootstrapping techniques, which iteratively tag unlabeled examples and add those with high confidence to the training set.
In the filtering process, two criteria must be considered: reliability and informativeness. We only retain sufficiently reliable instances, or the remainder will confuse the final model.

The informativeness of an instance is also important. We do not need additional instances if they are absolutely positive or negative. Deciding their labels is trivial for our initial classification model. In the terminology of SVM, they are not support vectors since they contribute nothing to the decision boundary in training. In testing, their output values by SVM are always greater than 1 or less then -1, which means they are distant from the separating hyperplane. Therefore, we can discard such

uninformative instances to reduce the size of the training set without diminishing performance.

Following these criteria, we now illustrate our filtration process. The flowchart of the whole procedure is shown in Figure 2. We use the initial model trained with TP+TN to label the LP data we collected. Those abstracts in the original LP with an SVM output in $[\gamma^+, 1]$ are retained. The dataset after filtering out irrelevant instances in LP is referred to as 'selected likely-positive data' (LP*).

The construction of selected likely-negative (LN*) data is similar. We collect 50k unlabeled abstracts from the PubMed biomedical literature database and classify them by our initial model. The articles with an SVM output in $[-1, \gamma^-]$ are collected into the LN* dataset.

The two thresholds γ^+ and γ^- are empirically determined to be 0 and -0.9, respectively. We use a looser threshold to filter LP data because of our prior knowledge of their reliability: after all, they have been recorded as PPI-relevant in some databases.

Exploiting likely positive and negative Data

The final issue is how to utilize these filtered instances. As shown in Figure 2, the likely data (LP* + LN*) are used to train a SVM model, the ancillary model, which is completely independent of the original training set. Subsequently, we use the ancillary model to predict all TP and TN instances, though their labels are already known, and these predicted values are scaled by a factor κ and encoded as additional features in the final model. In this manner, the final model can assign a suitable weight to the output of the ancillary model based on its accuracy in predicting the training set,

which is assumed to be close to the accuracy in predicting the test set. The scaling factor κ can be regarded as a prior confidence in the ancillary model.

References

1. Krallinger M, Valencia A: **Evaluating the Detection and Ranking of Protein Interaction Relevant Articles: the BioCreative Challenge Interaction Article Sub-task (IAS)**. In: *Second BioCreAtIvE Challenge Workshop: 2007*; 2007: 29-39.

2. Bader GD, Betel D, Hogue CW: **BIND: the Biomolecular Interaction Network Database**. *Nucleic Acids Res* 2003, **31**(1):248-250.

3. Xenarios I, Rice DW, Salwinski L, Baron MK, Marcotte EM, D E: **DIP: the database of interacting proteins**. *Nucleic Acids Res* 2000, **28**(1):289-291.

4. Cohen KB, Hunter L: **Natural Language Processing and Systems Biology**. In: *Artificial Intelligence and Systems Biology.* Edited by Dubitzky W, Azuaje F: Springer; 2005.

5. Donaldson I, Martin J, Bruijn Bd, Wolting C, Lay V, Tuekam B, Zhang S, Baskin B, Bader GD, Michalickova K *et al*: **PreBIND and Textomy – mining the biomedical literature for protein-protein interactions using a support vector machine**. *BMC Bioinformatics* 2003, **4**(11).

6. Marcotte EM, Xenarios I, Eisenberg D: **Mining literature for protein–protein interactions**. *Bioinformatics* 2001, **17**(4):359-363.

7. Lan M, Tan CL, Low H-B: **Proposing a New Term Weighting Scheme for Text Categorization**. In: *AAAI-06: 2006*; 2006.

8. Scott S, Matwin S: **Feature engineering for text classification**. In: *ICML-99: 1999*; 1999.

9. Paradis F, Nie J-Y: **Filtering Contents with Bigrams and Named Entities to Improve Text Classification**. In: *AIRS-05: 2005*; 2005.

10. Hermjakob H, Montecchi-Palazzi L, Lewington C, Mudali S, Kerrien S, Orchard S, Vingron M, Roechert B, Roepstorf P, Valencia A *et al*: **IntAct: an open source molecular interaction database**. *Nucleic Acids Res* 2004, **32**(Database issue):D452–D455.

11. Zanzoni A, Montecchi-Palazzi L, Quondam M, Ausiello G, Helmer-Citterich M, Cesareni G: **MINT: a Molecular INTeraction database**. *FEBS Lett* 2002, **513**(1):135-140.

12. **Peri S**, **Navarro JD**, **Amanchy R**, **Kristiansen TZ**, **Jonnalagadda CK**, **Surendranath V**, **Niranjan V**, **Muthusamy B**, **Gandhi TKB**, **Gronborg M** *et al*: **Development of Human Protein Reference Database as an Initial**

Platform for Approaching Systems Biology in Humans. *Genome Res* 2003, **13**:2363-2371.

13. Güldener U, Münsterkötter M, Oesterheld M, Pagel P, Ruepp A, Mewes H-W, Stümpflen V: **MPact: the MIPS protein interaction resource on yeast**. *Nucleic Acids Res* 2006, **34**(Database Issue):D436-D441.

14. Breitkreutz B-J, Stark C, Tyers M: **The GRID: the General Repository for Interaction Datasets**. *Genome Biol* 2003, **4**(3).

15. Moschitti A, Basili R: **Complex linguistic features for text classification: A comprehensive study.** . In: *ECIR-04: 2004*; 2004.

16. Joachims T: **Text Categorization with Support Vector Machines: Learning with Many Relevant Features**. In: *ECML-98: 1998*; 1998.

17. Cristianini N, Shawe-Taylor J: **An Introduction to Support Vector Machines**: Cambridge University Press; 2000.

18. Liu B, Lee WS, Yu PS, Li X: **Partially Supervised Classification of Text Documents** In: *Proceedings of the Nineteenth International Conference on Machine Learning (ICML-2002): 2002*; 2002.

Figures

Figure 1. A PPI record in the MINT database

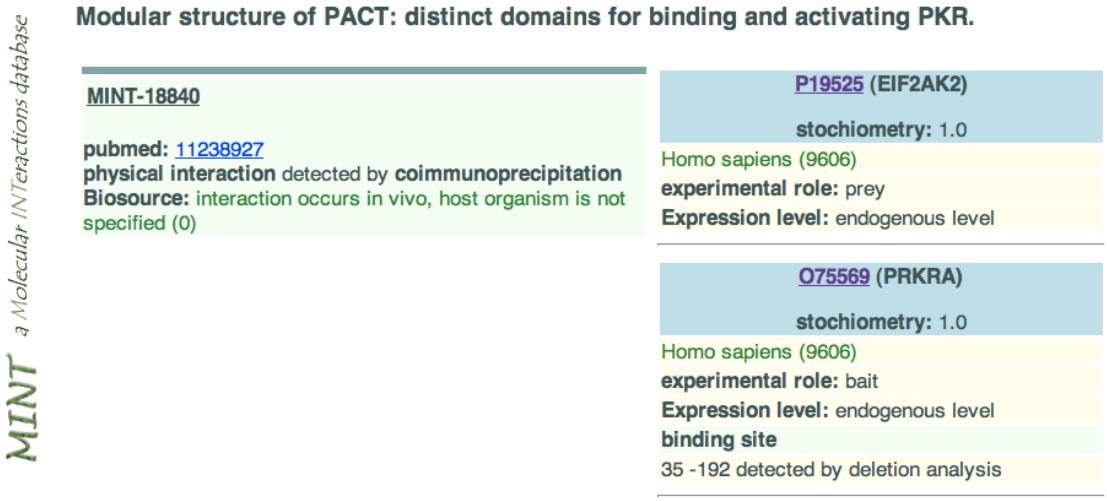
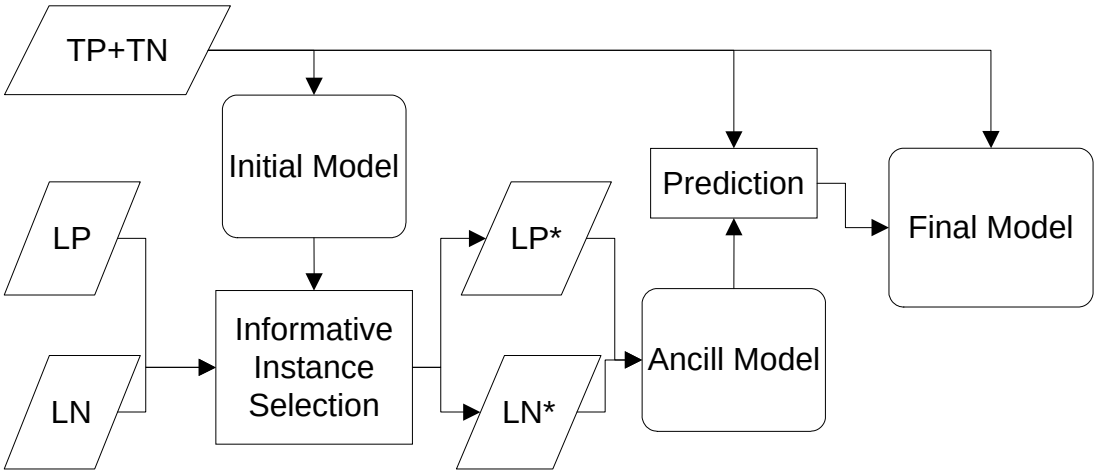


Figure 2. Flowchart of constructing the final model



Tables

Table 1. The contingency table for document frequency of term t_i in different classes. $\neg t_i$ stands for all words other than t_i

Class	t_i	$\neg t_i$
Positive	w	x
Negative	y	z

Table 2: Datasets used in our experiment

	Dataset	Size (# of abstracts)
Training	True Positive (TP)	3,536
	True Negative (TN)	1,959
	Likely Positive (LP)	18,930
	Likely Negative (LN)	105,000
Test	Positive	338
	Negative	339

Table 3. Scores of different methods using PNs, all scores are shown in percentage

Weight / Method	RF		TFRF		TFIDF	
	F	AUC	F	AUC	F	AUC
BoW	78.01	84.60	75.67	81.64	73.52	79.32
BoW + BoP	78.31	84.26	75.85	80.50	73.28	79.12
BoW + BoN	78.13	84.45	75.62	80.97	73.32	78.96
CBoW	**80.14**	**87.88**	**76.88**	**83.35**	**75.62**	**80.27**

Table 4. Scores of original training set vs. the expanded one, all scores are shown in percentage

Weight / Method	RF		TFRF		TFIDF	
	F	AUC	F	AUC	F	AUC
TN+TP	80.14	87.88	76.88	83.35	75.62	80.27
+LN*+LP*	**80.34**	**88.06**	**78.69**	**85.72**	**77.70**	**83.48**

Table 5. Compared with BioCreAtIvE-II systems

System	F(%)	AUC(%)
Our system	**80.34**	**88.06**
BioCreAtIvE-II best	78.00	85.54
BioCreAtIvE-II median	72.24	75.15

Table 6. Words with higher variance in weights assigned by SVM

Term	Bag 0	Bag 1	Bag 2	Dev.
interact	-0.031	0.150	0.294	0.163
bind	0.060	-0.017	0.236	0.130
vitro	-0.043	0.038	0.180	0.113
with	0.029	0.073	0.243	0.113
bound	-0.006	-0.011	0.168	0.102
association	-0.001	0.068	0.188	0.096
specifically	-0.037	0.055	0.150	0.093
interaction	0.243	0.227	0.393	0.092
identify	-0.045	0.081	0.124	0.088
localize	0.031	0.020	0.171	0.084
stimulation	-0.007	0.012	0.142	0.081
regulate	0.042	-0.008	0.147	0.079
complex	0.125	0.212	0.281	0.078
phosphorylation	-0.012	-0.007	0.124	0.077
target	-0.020	0.006	0.121	0.075

Table 7. Examples corrected by CBoW

Type	PMID	Content
FP→TN	9707401	In eukaryotes, assembly of the mitotic spindle requires the *interaction* of chromosomes *with* microtubules
FN→TP	16286467	We describe a mechanism whereby IL-1beta *stimulation* of NFkappaB is partially *regulated* by H_2O_2-mediated activation of NIK and subsequent NIK-mediated *phosphorylation* of IKKalpha

Table 8. Summary of weights in different bags

	Bag 0	Bag 1	Bag 2
Mean	-5.6×10^{-5}	60×10^{-5}	104×10^{-5}
Dev.	1.38×10^{-2}	1.39×10^{-2}	1.77×10^{-2}

Table 9. *p*-values of hypothesis test on the equality of means and standard deviations of weights

	Bag 0 vs. 1	Bag 0 vs. 2	Bag 1 vs. 2
Mean	5.813×10^{-6} *	2.642×10^{-7} *	0.2926
Dev.	0.3158	$<2.2\times10^{-16}$ *	$<2.2\times10^{-16}$ *

*significant difference