

PROCEEDINGS

Open Access

# Combining active learning and semi-supervised learning techniques to extract protein interaction sentences

Min Song<sup>1\*</sup>, Hwanjo Yu<sup>2†</sup>, Wook-Shin Han<sup>3†</sup>

From 9th International Workshop on Data Mining in Bioinformatics (BIOKDD)  
Washington, DC, USA. 25 July 2010

## Abstract

**Background:** Protein-protein interaction (PPI) extraction has been a focal point of many biomedical research and database curation tools. Both Active Learning and Semi-supervised SVMs have recently been applied to extract PPI automatically. In this paper, we explore combining the AL with the SSL to improve the performance of the PPI task.

**Methods:** We propose a novel PPI extraction technique called PPIspotter by combining Deterministic Annealing-based SSL and an AL technique to extract protein-protein interaction. In addition, we extract a comprehensive set of features from MEDLINE records by Natural Language Processing (NLP) techniques, which further improve the SVM classifiers. In our feature selection technique, syntactic, semantic, and lexical properties of text are incorporated into feature selection that boosts the system performance significantly.

**Results:** By conducting experiments with three different PPI corpuses, we show that PPIspotter is superior to the other techniques incorporated into semi-supervised SVMs such as Random Sampling, Clustering, and Transductive SVMs by precision, recall, and F-measure.

**Conclusions:** Our system is a novel, state-of-the-art technique for efficiently extracting protein-protein interaction pairs.

## Background

Automated protein-protein interaction (PPI) extraction from unstructured text collections is a task of significant interest in the bio-literature mining field. The most commonly addressed problem has been the extraction of binary interactions, where the system identifies which protein pairs in a sentence have a biologically relevant relationship between them. Proposed solutions include both hand-crafted rule-based systems and machine learning approaches [1]. Recently Semi-supervised Learning (SSL) techniques have been applied to PPI tasks [2]. SSL is a Machine Learning (ML) approach

that combines supervised and unsupervised learning where typically a small amount of labeled and a large amount of unlabeled data are used for training. SSL has gained significant attention to PPI extraction because of two reasons. First, labeling of a large set of instances is labor-intensive and time-consuming. This task has to be also carried out by qualified experts and thus is expensive. Second, several studies show that using unlabeled data for learning improves the accuracy of classifiers [3,4].

One major problem of SSL is that it may introduce incorrect labels to the training data, as the labeling is done by machine, and such labeling errors are critical to the classification performance. Active Learning (AL) can complement the SSL by reducing such labeling errors [5]. AL is a technique of selecting a small sample from the unlabeled data such that labeling on the sample

\* Correspondence: min.song@njit.edu

† Contributed equally

<sup>1</sup>Information Systems Department, New Jersey Institute of Technology, University Heights, Newark, New Jersey, USA

Full list of author information is available at the end of the article

maximizes the learning accuracy. The selected sample is manually labeled by experts. In this paper, we explore combining the AL with the SSL to improve the performance of the PPI task. To our best knowledge, this is the first attempt to apply a combination of semi-supervised and active learning for the extraction task of protein-protein interaction.

The contributions of this paper are three fold. First, we proposed a novel PPI extraction technique called PPISpotter by combining Deterministic Annealing-based SSL and an AL technique to extract protein-protein interaction. Second, we extracted a comprehensive set of features from MEDLINE records by Natural Language Processing (NLP) techniques, which further improve the SVM classifiers. In our feature selection technique, syntactic, semantic, and lexical properties of text are incorporated into feature selection that boosts the system performance significantly. Third, we conducted experiments with three different PPI corpuses and showed that PPISpotter is superior to other techniques by precision, recall, and F-measure.

Many approaches have been proposed to extract protein-protein interaction from unstructured text. One approach employs pre-specified patterns and rules for PPI extraction [6]. However, this approach is often inapplicable to complex cases not covered by the pre-defined patterns and rules. Huang et al. [7] proposed a method where patterns are discovered automatically from a set of sentences by dynamic programming.

The second approach utilizes dictionary. Blaschke et al. [8] extracted protein-protein interactions based on co-occurrence of the form "... p1...I1... p2" within a sentence, where p1, p2 are proteins and I1 is an interaction term. Protein names and interaction terms (e.g., activate, bind, inhibit) are provided as a "dictionary." Pustejovsky et al. [9] extracted an "inhibit" relation for the gene entity from MEDLINE. Jenssen et al. [10] extracted gene-gene relations based on co-occurrence of the form "... g1...g2..." within a MEDLINE abstracts, where g1 and g2 are gene names. Gene names were provided as a "dictionary", harvested from HUGO, Locus-Link, and other sources. Although their study uses 13,712 named human genes and millions of MEDLINE abstracts, no extensive quantitative results are reported and analyzed. Friedman et al. [11] extracted a pathway relation for various biological entities from a variety of articles.

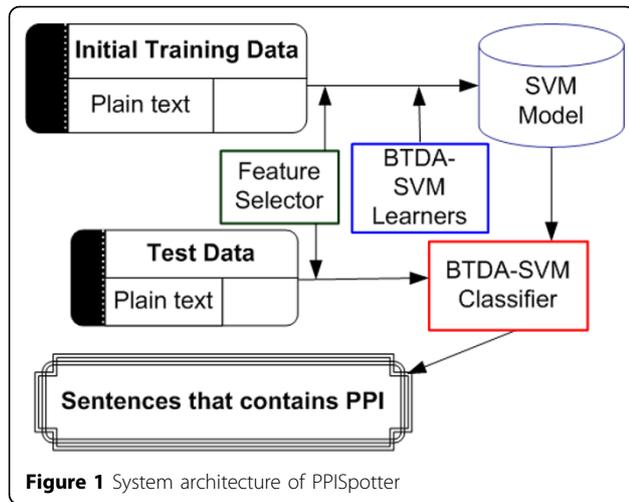
The third approach is based on machine learning techniques. Bunescu et al. [1] conducted protein/protein interaction identification with several learning methods such as pattern matching rule induction (RAPIER), boosted wrapper induction (BWI), and extraction using longest common subsequences (ELCS). ELCS automatically learns rules for extracting protein interactions

using a bottom-up approach. They conducted experiments in two ways; one with manually crafted protein names and the other with the extracted protein names by their name identification method. In both experiments, Zhou et al. [12] proposed two novel semi-supervised learning approaches, one based on classification and the other based on expectation-maximization, to train the HVS model from both annotated and un-annotated corpora. Song et al. [13] utilized syntactical, as well as semantic cues, of input sentences. By combining the text chunking technique and Mixture Hidden Markov Models, They took advantage of sentence structures and patterns embedded in plain English sentences. Temkin and Gilder [14] used a full parser with a lexical analyzer and a context free grammar (CFG) to extract protein-protein interaction from text. Alternatively, Yakushiji et al. [15] propose a system based on head-driven phrase structure grammar (HPSG). In their system protein interaction expressions are presented as predicate argument structure patterns from the HPSG parser. These parsing approaches consider only syntactic properties of the sentences and do not take into account semantic properties. Thus, although they are complicated and require many resources, their performance is not satisfactory. Mitsumori et al. [16] used SVM to extract protein-protein interactions. They use bag-of-words features, specifically the words around the protein names. These systems do not use any syntactic or semantic information. Miyao et al. [17] conducted a comparative evaluation of several state-of-the-art natural language parsers, focusing on the task of extracting protein-protein interaction (PPI) from biomedical papers. They found marginal difference in terms of accuracy but more significant differences in parsing speed. BioPPISV-MExtractor is a recent PPI extraction system developed with SVM [2]. It utilizes rich feature sets such as word features, keyword feature, protein names distance feature, and Link Grammar extraction results for protein-protein interaction extraction. They observed that the rich feature sets help improve recall at the cost of a moderate decline in precision.

Cui et al. [18] applied an uncertainty sampling based method of active learning for a lexical feature-based SVM model to tag the most informative unlabeled samples. They reported that the performance of the active learning-based technique on AIMED and CB corpora was significantly improved in terms of reduction of labelling cost.

## Methods

In this section, we describe the overall architecture and procedures of PPISpotter (Figure 1). PPISpotter incorporates AL models into SSL SVMs for extraction of protein-protein interaction. PPISpotter also automatically

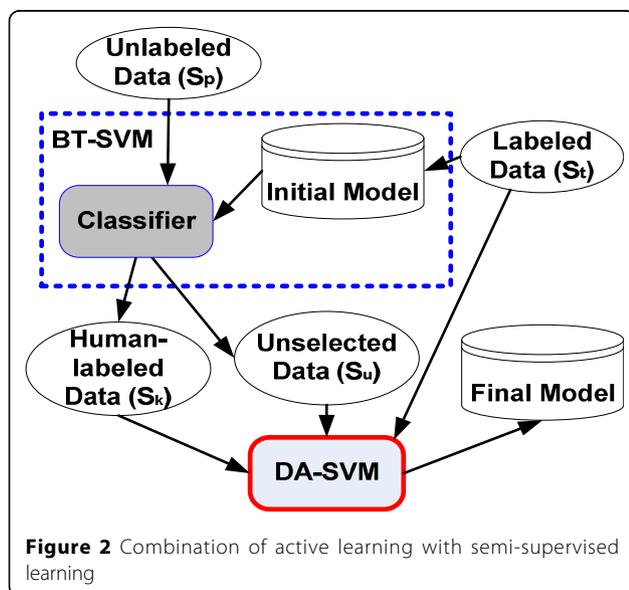


converts a sentence into 9 feature sets based on the technique described in Section 4.

Below is a set of steps that PPIspotter processes.

**Step 1:** Preprocess the initial training data. The feature selector applies the feature selection technique proposed in Section 4 to the preprocessed data sets.

**Step 2:** Train the model. Two classifiers, Break Tie-based SVM (BT-SVM) and Deterministic Annealing-based SVM (DA-SVM) classifiers are combined to train the model (a.k.a. BTDA-SVM). Figure 2 illustrates how to combine these two techniques (Blue dot line is the BT-SVM procedure and red solid line is the DA-SVM procedure). At this stage, the human expert provides feedback to the system for a set of instances in the fuzzy unlabeled data. Note that the BT-SVM classifier is based on the Break Tie active learning approach and



DA-SVM classifier is based on the Deterministic Annealing technique.

**Step 3:** Take the input data and convert it to the same format as the training data. The feature selector performs the same task as in Step 1.

**Step 4:** Apply the BTDA-SVM learner to identify sentences that contain protein-protein interaction.

**Step 5:** Store extracted sentences to the database.

### Combination of active learning with semi-supervised learning

One of the goals of this paper is to combine SSL and AL into a unified semi-supervised active learning technique for protein-protein interaction extraction. We employ a proportion of unlabeled data in the learning tasks in order to resolve the problem of insufficient training data.

Our strategy of combining AL with SSL is inspired by the Tur et al.'s study [5]. We employ the break tie AL technique (BT-SVM) to train a classifier on both labeled and unlabeled data, and return to the user the most relevant results. Then, the learning system trains a classifier based on the Deterministic Annealing SSL technique (DA-SVM) on both the labeled and unlabeled data ( $S_v$ ,  $S_k$ , and  $S_u$ ), and results in the final model (Figure 2).

BTDA-SVM is a combination of the active learning algorithm presented in Section 4 and the semi-supervised learning algorithm presented in Section 5. Instead of leaving out the instances classified with high confidence scores, this algorithm exploits them. Figure 3 explains the BTDA-SVM algorithm.

### Active learning

Active learning, known as pool-based active learning, is an interactive learning technique designed to reduce the labor cost of labeling in which the learning algorithm can freely assign the unlabeled data instances to the training set. The basic idea is to select the most informative data instances for labeling by the users in the next learning round. In other words, the strategy of active learning is to select an optimal set of unlabeled data instances that minimizes the expected risk of the next round.

#### Breaking tie (BT)

For a given instance, the regular SVMs results in distances among instances whose range is from 0 to 1. The value 0 means that the instance lies on the hyperplane and the value 1 indicates that the instance is a support vector.

To assign a probability value to a class the sigmoid function can be used with the assumption that a probability associated with a classifier indicates to which extent the classification result is trusted. In this case,

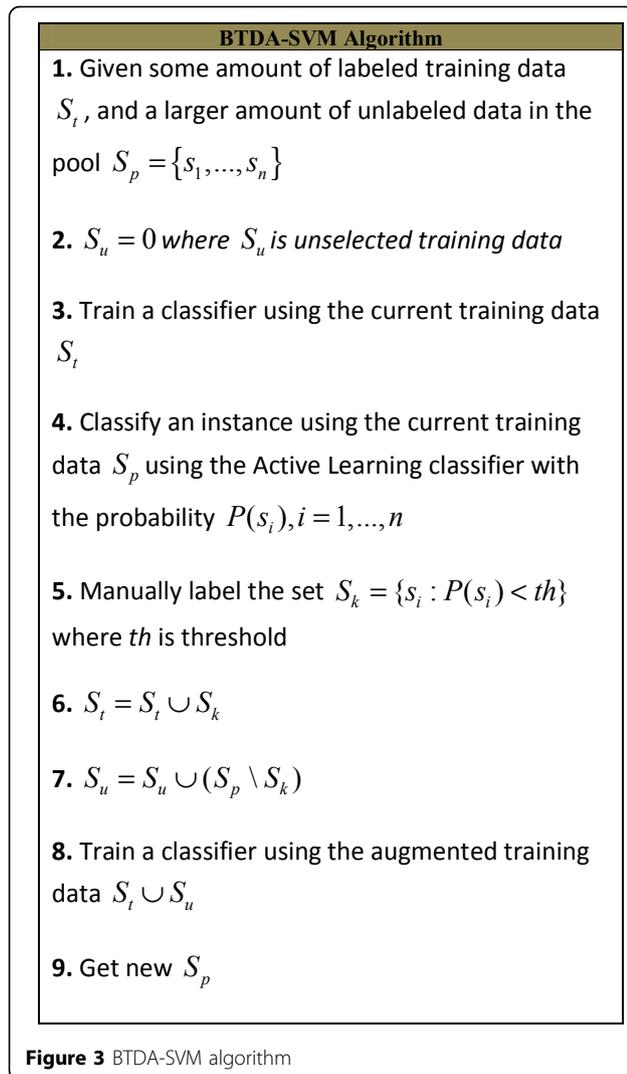


Figure 3 BTDA-SVM algorithm

Luo et al. [19] defines the parametric model in the following form:

$$P(y = 1 | f) = \frac{1}{1 + \exp(Af + B)} \quad (1)$$

where  $A$  and  $B$  are scalar values, which have to be estimated and  $f$  is the decision function of the SVMs. This parametric model is used for calculating the probabilities. To use this model, the SVM parameters (complexity parameter  $C$ , kernel parameter  $k$ ) and the parameter  $A$  and  $B$  need to be calculated. Although cross validation can be used for this calculation, it is computationally expensive. An alternative is a pragmatic approximation method that all binary SVMs have the same  $A$  while eliminating  $B$  by assigning 0.5 to instances lying on the decision boundary and by trying to compute the SVM parameters and  $A$  simultaneously [19].

The decision function can be normalized by its margin to include the margin in the calculation of the probabilities.

$$P_{pq}(y = 1 | f) = \frac{1}{1 + \exp\left(\frac{Af}{\|\omega\|}\right)} \quad (2)$$

where we currently look at class  $p$  and  $P_{pq}$  is the probability of class  $p$  versus class  $q$ . We assume that  $P_{pq}, q=1,2,\dots$  are independent. The final probability for class  $p$ :

$$P(p) = \prod_q^{q \neq p} P_{pq}(y = 1 | f) \quad (3)$$

It has been reported that the performance bases on this approximation is fast and accurate [19]. This probability model serves as basis for the Breaking Tie algorithm for semi-supervised learning.

#### Semi-supervised support vector machines

Support Vector Machines (SVMs) is a supervised machine learning approach designed for solving two-class pattern recognition problems. SVMs adopts maximum margin to find the decision surface that separates the positive and negative labeled training examples of a class [20].

Transductive Support Vector Machines (TSVMs) is an extended version of SVM that uses unlabeled data in addition to labeled data for train classifiers [21]. The goal of TSVMs is to determine which test data instances result in the maximum-margin hyperplane that separates the positive and negative examples for classifiers. Since every test instances need to be included in the SVM's objective function, finding the exact solution to the resulting optimization problem is intractable. To resolve this issue, Joachims [21] proposed an approximation algorithm. One issue of Joachims' approach, however, is that it requires the similar distribution of positive and negative instances between the test data and the training data. This requirement is difficult to meet particularly when the training data is small. The challenge is to find a decision surface that separates the positive and negative instances of the original labeled data and the unlabeled data to unlabeled data to be converted to labeled data with maximum margin. The unlabeled data sets apart the decision boundary from the dense regions, and the optimization problem is NP-hard [22]. Various approximation algorithms are found in [22].

The optimization problem held in TSVMs is a non-linear non-convex optimization [23]. Past several years, researchers have attempted to solve this critical

problem. Chapell and Zien [24] proposed a smooth loss function, and a gradient descent method to find the decision boundary in a region of low density. Another technique is a branch-and-bound method [25] that searches for the optimal solution. But, it is applicable to a small number of examples due to involving the heavy computational cost. Despite the success of TSVM, the unlabeled data does not necessarily improve classification accuracy.

As an alternative to TSVMs, we explore an Deterministic Annealing approach to semi-supervised SVMs. The first approach was proposed by Luo and his colleagues [19] that formulated a probabilistic framework for image recognition. The Deterministic Annealing (DA) approach is the second proposed by Sindhvani et al. [26]. In the probabilistic framework, semi-supervised learning can be modeled as a missing data problem, which can be addressed by generative models such as mixture models. In the case of semi-supervised learning, probabilistic approaches provide us with various different ways to query unlabeled instances for labeling. A simple method is to train a model on the given labeled datasets and use this model on the unlabeled data. Each of these unlabeled instances is given probabilities that these instances belong to a given class. We can query the least certain instances or the most certain instances. The detailed description of the Deterministic Annealing semi-supervised learning is provided in the study of Luo and his colleagues [19].

**Deterministic annealing (DA)**

Deterministic annealing (DA) is a special case of a homotopy method for combinatorial optimization problems [26]. We adopt the DA technique proposed by Sindhvani et al. [26] to extraction of protein-protein interaction. The detailed description of applying DA for SVMs is provided by Sindhvani et al. [26].

Suppose one is given a following non-convex optimization problem:

$$\gamma^* = \arg \min_{\gamma \in \{0,1\}^n} F(\gamma) \tag{4}$$

DA finds a local minimum of this in the following: First, DA treats the discrete variables as random binary variables over a space of probability distributions  $P$ . Second, to solve the optimization problem, DA finds a distribution  $p \in P$  that minimizes the expected value of  $F$ . It makes the optimization problem to be continuous. For this reason, an additional convex term is added to the objective function which is the entropy  $S$  of the distribution denoted in Eq. 1.

$$p^* = \arg \min_{p \in P} E_p(F(\gamma)) - T.S(p) \tag{5}$$

where the parameter  $T$  controls the trade-off between the expectation and the entropy (called the temperature of the problem) and  $\gamma \in \{0,1\}^n$  are the discrete variables for the objective function  $F(\gamma)$ . For  $T = 0$  and  $P$  including all point-mass distributions over  $\{0,1\}^n$ , the global minimizer  $P^*$  in Eq. 1 will place all of its mass on the global minimizer of  $F$ . However, if  $T \gg 0$ , the entropy term in Eq. (1) dominates the objective function. With convexity, we can solve a sequence of problems for values of  $T_0 > T_1 > \dots > T_\infty = 0$  where each of them is initialized at the solution obtained by the previous one. This sequence of temperatures is called as the annealing schedule. When  $T$  is close to zero the influence of the entropy term becomes shrunken. Therefore, the distribution becomes more concentrated on the minimum of  $E_p[F]$  which allows us to identify the discrete variables  $\gamma$  by  $p$ . Note that there is no guarantee for global optimality because there is not always a path connecting the local minimizers for the chosen sequence of  $T$  to the global optimum of  $F$ .

**Applying DA to SVMs**

Given a binary classification problem, we consider a set of  $L$  training pairs  $L = \{(x_1, y_1), \dots, (x_L, y_L)\}$ ,  $x \in \mathbb{R}^n$ ,  $y \in \{1, -1\}$  and an unlabeled set of  $U$  test vectors  $U = \{x_{L+1}, \dots, x_{L+U}\}$  SVMs have a decision function  $f_\theta(\cdot)$  of the form  $f_\theta(x) = w \cdot \Phi(x) + b$ , where  $\theta = (w, b)$  are the parameters of the model, and  $\Phi(\cdot)$  is the chosen feature map, often implemented implicitly using the kernel trick. Given a training set  $L$  and a test set  $U$ , for the TSVM optimization problem, find among the possible binary vectors  $\{\gamma = (y_{L+1}, \dots, y_{L+U})\}$  the one such that an SVM trained on  $L \cup (U \times \gamma)$  yields the largest margin. This combinatorial problem can be approximated as finding an SVM separating the training set under constraints which force the unlabeled examples to be as far as possible from the margin. This can be written as minimizing

$$\frac{1}{2} \|w\|^2 + C \sum_{i=1}^L \xi_i + C^* \sum_{i=L+1}^{L+U} \xi_i \quad \text{subject to}$$

$$|f_\theta(x_i)| \geq 1 - \xi_i, i = L + 1, \dots, L + U \text{ and}$$

$$|f_\theta(x_i)| \geq 1 - \xi_i, i = L + 1, \dots, L + U.$$

This minimization problem is equivalent to minimizing

$$w^* = \min_{w, \{\gamma_j\}_{j=1}^{L+U}} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^L H_1(y_i f_\theta(x_i)) + C^* \sum_{i=L+1}^{L+U} H_1(|f_\theta(x_i)|) \tag{6}$$

where the function  $H_1(\cdot) = \max(0, 1 - \cdot)$  is the classical Hinge Loss function. In other words, TSVM seeks a hyperplane  $w$  and a labeling of the unlabeled examples, so that the SVM objective function is minimized. The discussion in Deterministic Annealing motivates a continuous objective function,

$$\tau_T(f, p) = E_p \tau(f, \gamma') - TS(p) \cdot \quad (7)$$

that defined by taking the expectation of  $\tau(f, \gamma')$  (Eq. 1) with respect to a distribution  $p$  on  $\gamma'$  and including entropy of  $p$  as a homotopy term.

For a fixed  $T$ , the solution to the optimization problem above is tracked as the temperature parameter  $T$  is lowered to 0. The DA algorithm returns the solution corresponding to the minimum value achieved when some stopping criterion is satisfied. The criterion used in the DA algorithm is the Pair-wise Mutual Information (PMI) between values of  $p$  in consecutive iterations. The parameter  $T$  is decreased in an outer loop until the total entropy falls below a threshold.

### Feature selection

Rich feature sets improve accuracy of the PPI extraction task [27]. The features used in Yang's paper include word features, keyword features, protein name distance features, and link path features, etc. In this paper, we explore various different features such as syntactic and lexical features as well as semantic features such as negated sentence features, interactor and its POS tag features into the feature sets. The total 9 features were selected for our semi-supervised learning technique (See Table 1).

**Negation:** We include whether a sentence is negated or not in the feature set. We use NegEx developed by Chapman and colleagues [28] for negation. NegEx is a regular expression-based approach that defines a fairly extensive list of negation phrases that appear before or after a finding of negation. NegEx treats a phrase as a negated one if a negation phrase appears within  $n$  words of a finding. The output of NegEx is the negation status assigned to each of the UMLS terms identified in the sentence: negated, possible or actual. NegEx uses the following regular expressions triggered by three types of negation phrases:

*<pre-UMLS negation phrase> {0-5 tokens} <UMLS term> and <UMLS term> {0-5 tokens} <post-UMLS negation phrase>*

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

Feature	Feature Value
Is negated sentence	True
No. of protein occurrences	3
Interactor name	response
Interactor POS	NN
Interactor position	88
No. of words in between proteins	24
No. of left words	-1
No. of right words	12
Link path status	Yes

There are three types of negation phrases in these expressions: 1) pre-UMLS, 2) post-UMLS and 3) pseudo negation phrases. Pre-UMLS phrases appear before the term they negate, while the post-UMLS phrases appear after the term they negate. Pseudo negation phrases are similar with negation phrases but are not reliable indicators of negation; they are used to limit the negation scope. All UMLS terms inside of the 0-5 tokens window are assigned the negation status depending on the nature of the negation phrase: negated or possible. The example of the negated sentence processed with NegEx is as follows:

*[PREN].No[PREN] relevant changes in heart rate , body weight , and plasma levels of [NEGATED]renin [NEGATED] activity and aldosterone concentration were observed → negated*

**Number of proteins named entities (NE) occurrences:** We extracted protein names from each sentence by using a Conditional Random Field (CRF)-based Named Entity Recognition (NER) technique.

To train the CRF NER, we used the training data provided for the BioCreative II Gene Mention Tagging task. The training data consist of 20,000 sentences. Approximately 44,500 GENE and ALTGENE annotations were converted to the MedTag database format [29]. Once we built the train model, we applied the CRF NER to extract proteins or genes from a sentence and counted the number of occurrences of genes in the sentence.

**Interactor:** Interactor is the term that shows the interaction among proteins in a sentence. The total of 220 interactor terms was identified. We applied a modified UEA stemmer to take care of term variations of interactor [30]. We did not apply an aggressive stemmer like Porter stemmer since we wanted to preserve the POS tag of the interactor.

**Interactor POS:** As for protein named entities, we applied the CRF-based POS tagging technique to tag tokenized words in a sentence. The CRF-based POS tagger was built on top of the MALLET package [31].

**Interactor position:** We included the position of the interactor term in a sentence in the feature set.

**Number of words in between proteins:** We included the number of words in a left most Protein NE and a right most Protein NE in the feature set.

**Number of left words:** We included the number of words in the left side of the first appearance of a Protein NE in the feature set.

**Number of right words:** We included the number of words in the right side of the last appearance of Protein NE in the feature set.

**Link path status:** This feature set is obtained by Link Grammar that was introduced by Lafferty et al. [32]. Link Grammar is used to connect pairs of words in a sentence with various links. Each word is linked with

connectors. A link consists of a left-pointing connector connected with a right-pointing connector of the same type on another word. A sentence is validated if all the words are connected. We assume that if a link path between two protein names exists, these two proteins have interaction relation. In our feature selection, if a Link path between two protein names exists, it is set to “Yes”, otherwise, “No”. The Link Grammar parser was used in several papers to extract protein-protein interaction [27,2].

## Results

### Data sets

One of the issues in protein-protein interaction extraction is that different studies use different data sets and evaluation metrics. It makes it difficult to compare the results reported from the studies.

In this paper, we used three different datasets that have been widely used in protein-protein interaction tasks. These are 1) the AIMED corpus, 2) the BioCreAtIvE2 corpus that is provided as a resource by BioCreAtIvE II (Critical Assessment for Information Extraction in Biology) challenge evaluation, and 3) BioInfer corpus. Table 2 summarizes the characteristics of these three datasets.

**AIMED:** Bunescu et al. [1] manually developed the AIMED corpus3 for protein-protein interaction and protein name recognition. They tagged 199 Medline abstracts, obtained from the Database of Interacting Proteins (DIP) and known to contain protein interactions. This corpus is becoming a standard, as it has been used in the recent studies in several studies [1,15,16].

**BioCreAtIvE2:** is a corpus for protein-protein interactions, originated from the BioCreAtIvE task 1A data set for named entity recognition of gene/protein names. We randomly selected 1000 sentences from this set and added additional annotation for interactions between genes/proteins. 173 sentences contain at least one interaction, 589 sentences contain at least one gene/protein. There are 255 interactions, some of which include more than two partners (e.g., one partner occurs with full name and abbreviated) [33].

**BioInfer:** stands for Bio Information Extraction Resource. It was developed by Pyysalo et al. [34]. The corpus contains 1100 sentences from PubMed abstracts

annotated for relationships, named entities, as well as syntactic dependencies.

Since previous studies that used these datasets performed 10-fold cross-validation, we also performed 10-fold cross-validation in these datasets and reported the average results over the runs.

For evaluation methodology, we use precision, recall, F-score, and AUC as our metrics to evaluate the performances of the methods.

### Comparison techniques

In this section, we briefly describe other techniques incorporated into semi-supervised SVMs and used to evaluate the performance of active semi-supervised learning models adopted in PPISpotter.

#### Baseline: random sampling (RS-SVM)

Random sampling of the unlabeled instances is a naïve approach to semi-supervised learning. We use this approach to compare with the other semi-supervised learning approaches as several studies used this approach to compare it with other semi-supervised learning approaches [19,35].

#### Clustering (C-SVM)

One technique is a clustering algorithm applied for the unlabeled data. Fung and Mangasarian [19] used the k-median clustering and showed that the performance was competitive comparing to a supervised learning. The downside of a clustering approach is the correct number of the clusters needs to be pre-defined. We initially tried the two clustering techniques: K-means and Kernel K-means and found that there was only marginal difference in terms of performance. Therefore, we use K-means for the performance comparison.

#### Supervised SVMs (SVM)

The kernel we used as the baseline supervised SVM model is a linear kernel. One of the advantages of supervised SVMs with a linear kernel is that it can handle high dimensional data effectively. The reason is it compares the “active” features rather than the complete dimensions. This way, we can impose richer feature sets upon each training example to enhance system performance. The richer feature sets showed to be more effective than the simple feature sets [2]. Another advantage of linear kernel SVM is its low training and testing time costs. In addition, using linear kernel SVM only penalty parameter  $C$  needs to be adjusted in the algorithm, which is usually set as a constant in applications. In our experiments, the SVM-light package was used. The penalty parameter  $C$  in setting the SVM is an important parameter since it controls the tradeoff between the training error and the margin. The SVM-light package

**Table 2 Data sets used for experiments**

Data Set	Total Sentences	Positive Sentences	Negative Sentences
AIMED	4026	951	3075
BioCreative2	4056	2202	1854
BioInfer	1100	573	527

does an excellent job on setting the default value for this parameter. In our experiments the parameter was left as default value since we observed that other manually determined values of this parameter in fact led to worse performance of supervised SVMs when compared with the default one.

## Discussion

We evaluate and compare the performance of the active semi-supervised machine learning approach (BTDA-SVM) in several different ways. First, we compare it with three different techniques: random sampling, K-means clustering, and supervised SVMs. In addition, we test the performance of BTDA-SVM with supervised counterparts (SVMs) as well as an active learning technique (BT-SVM) for the task of protein-protein interaction extraction. Second, we exam whether the size of combined training datasets between unlabeled and labeled data have impact on the performance. As discussed in Section 3, we Break Tie and Deterministic Annealing, as a kernel function in BTDA-SVM.

Table 3 shows the results obtained with the AIMED data set. Our approach (BTDA-SVM) performs considerably better than other techniques in terms of precision, recall, and F-measure. BTDA-SVM's performance is superior to the regular SVMs approach by 34.79% in terms of precision. It is 25.55% better than the Random Sampling approach (RS-SVM) in terms of recall. In terms of F-measure, BTDA-SVM is 28.6% better than the regular SVMs. The Break Tie approach (BT-SVM) is the second best in terms of three measures.

We conducted individual t-tests essentially as specific comparisons. Our prediction that BTDA-SVM would be better than the other comparison techniques (BT-SVM, SVM, RS-SVM, and C-SVM) was confirmed  $t(11) = 3.6966E-11$ ,  $p < 0.05$  (one-tailed) at  $n-1$  degrees of freedom (12 runs) while comparing with C-SVM which performed best over the other two comparison techniques. Similarly, the t-test confirmed that the performance difference of BT-SVM is statistically significant from C-SVM  $t(11) = 0.0169$ ,  $p < 0.05$  (one-tailed).

**Table 3 Experimental results – AIMED data set**

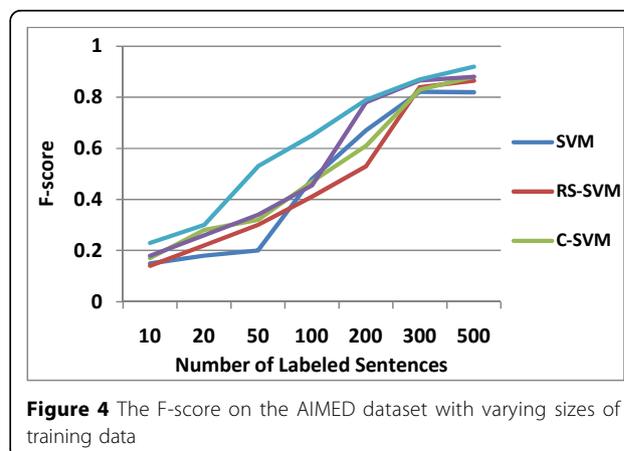
Algorithms	Measures		
	Precision	Recall	F-score
SVM	55.15%	42.47%	48.14%
RS-SVM	56.98%	41.71%	48.92%
C-SVM	64.53%	40.42%	50.67%
BT-SVM	65.23%	42.51%	53.64%
BTDA-SVM	74.34%	50.75%	61.91%
Yakushiji et al. [15]	33.70%	33.10%	33.40%
Mitsumori et al. [16]	54.20%	42.60%	47.70%

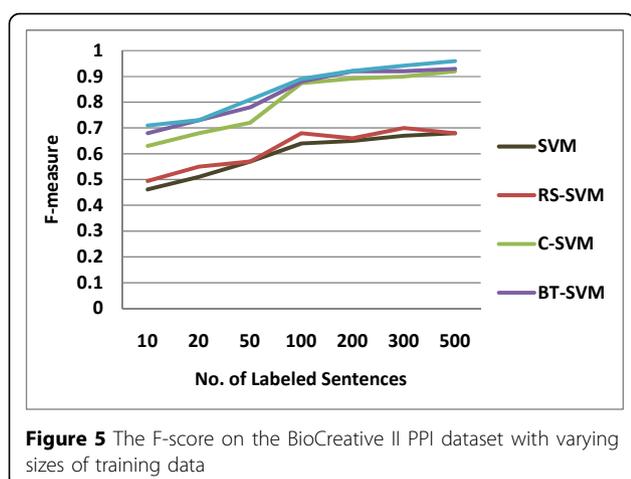
In Table 3, we also show the results obtained previously in the literature by using the same data set. Yakushiji et al. [15] used an HPSG parser to produce predicate argument structures. They utilized these structures to automatically construct protein interaction extraction rules. Mitsumori et al. [16] used SVMs with the unparsed text around the protein names as features to extract protein interaction sentences.

Semi-supervised approaches are usually claimed to be more effective when there is less labeled data than unlabeled data, which is usually the case in real applications. To see the effect of semi-supervised approaches we perform experiments by varying the amount of labeled training sentences in the range [10, 3000]. For each labeled training set size, sentences are selected randomly among all the sentences, and the remaining sentences are used as the unlabeled test set. The results that we report are the averages over 10 such random runs for each labeled training set size. We report the results for the algorithms when edit distance based similarity is used, as it mostly performs better than cosine similarity.

Figure 4 and 5 show the performance differences of five SVM-based learning techniques as the size of training data increases. BTDA-SVM performs considerably better than their supervised counterpart SVM, RS-SVM, C-SVM when we have small number of labeled training data. It is interesting to note that, although SVM is one of the best performing algorithms with more training data, it is the worst performing algorithm with small amount of labeled training sentences. Its performance starts to increase when number of training data is larger than 200. Eventually, its performance gets close to that of the other algorithms. Harmonic function is the best performing algorithm when we have less than 200 labeled training data.

BTDA-SVM consistently outperforms other techniques in this experiment. We observed that most of the techniques made significant improvement when the





training data reaches 200 training instances. Compared to other techniques, BTDA-SVM did not make a radical change to the size of training data.

Table 4 shows the experimental results with the BioCreative2 PPI data set. The performance with BTDA-SVM is always better than other techniques by three measures. BTDA-SVM outperforms the regular SVMs (SVM) by 22.34%, 86.13%, and 48.89% respectively in terms of precision, recall, and F-measure. The second best performance is achieved by BT-SVM in terms of three measures.

In t-test, we predict that BTDA-SVM would be better than the other three comparison techniques (SVM, RS-SVM, and C-SVM), and the prediction was confirmed  $t(11)=0.0312$ ,  $p<0.05$  (one-tailed) at  $n-1$  degrees of freedom (12 runs) while comparing with C-SVM. However, our prediction that BT-SVM would be better than C-SVM was not confirmed  $t(11)=0.092$ ,  $p<0.05$  (one-tailed). As shown in Figure 5, performance curves are different from ones with the AIMED data set. The performance of SVM and RS-SVM is consistently inferior to C-SVM, BT-SVM, and BTDA-SVM.

Although BTDA-SVM consistently outperforms other techniques in this experiment, it does not show statistical significance (In t-test,  $t(6)=0.2124$ ,  $p<0.05$  (one-tailed) at  $n-1$  degrees of freedom). In addition, all

techniques did not make a radical change to the size of training data.

We reported the performance of five comparison techniques with the BioInfer data set. Table 5 shows the experimental results in terms of precision, recall, and AUC. BTDA-SVM's performance is the best over the other four techniques. It is better than the regular SVMs approach by 25.23%, 19.41%, and 10.32% in terms of precision, recall, and AUC respectively. With respect to AUC, the results of the t-test indicates that BTDA-SVM's performance is statistically significantly better than the other three comparison techniques (SVM, RS-SVM, and C-SVM),  $t(11)=8.3483E-6$ ,  $p<0.05$  (one-tailed) at  $n-1$  degrees of freedom (12 runs) while comparing with C-SVM which performed best over the other two comparison techniques. In the same vein, our prediction that BT-SVM would be better than C-SVM was confirmed  $t(11)=0.00025$ ,  $p<0.05$  (one-tailed).

## Conclusions

The goal of our study is two-fold: The first is to explore integrating an active learning technique with semi-supervised SVMs to improve the performance of classifiers. The second is to propose rich, comprehensive feature sets for the protein-protein interaction. To this end, we presented an active semi-supervised SVM-based PPI extraction system, PPISpotter, which encompasses the entire procedure of PPI extraction from the biomedical literature: protein name recognition, rich feature selection, and PPI extraction. In PPI extraction stage, besides several common features such as word features and keyword features, some new useful features including protein names distance feature, phrase negation, and link path feature were introduced for the supervised learning problem. We combined an active learning technique, Break Tie (BT-SVM) with the Deterministic Annealing-based semi-supervised learning technique (DA-SVM), which serves the core algorithm for the PPISpotter system (BTDA-SVM). This BTDA-SVM technique, compared with four different techniques including an active learning technique (BT-SVM), was tested on three widely used PPI corpora. The experimental results indicated that our technique, BTDA-

**Table 4 Experimental results – BioCreative2 PPI data set**

Algorithms	Measures		
	Precision	Recall	F-score
SVM	70.23%	51.21%	58.33%
RS-SVM	71.7%	56.54%	62.5%
C-SVM	78.23%	88.68%	83.65%
BT-SVM	81.75%	93.5%	85.96%
BTDA-SVM	85.92%	95.32%	86.85%
TSVM-edit [36]	85.62%	84.89%	85.22%

**Table 5 Comparison results – BioInfer data set**

Algorithms	Measures		
	Precision	Recall	AUC
SVM	65.89%	54.6%	0.843
RS-SVM	64.5%	55.2%	0.847
C-SVM	70.24%	60.2%	0.86
BT-SVM	79.29%	63.1%	0.918
BTDA-SVM	82.52%	65.2%	0.93
Graph Kernel [1]	47.7%	59.9%	0.849

SVM, achieves statistically significant improvement over the other three techniques in terms of precision, recall, F-measure, and AUC.

In future work, we plan to further explore the characteristics of active learning approaches to semi-supervised SVMs and refine our approach to achieve a better PPI extraction performance.

#### Acknowledgements

Partial support for this research was provided by the National Science Foundation under grant DUE-0937629 and by the New Jersey Institute of Technology. This work was also supported by the Brain Korea 21 Project in 2010 and Mid-career Researcher Program through NRF grant funded by the MEST (No. KRF-2009-0080667).

This article has been published as part of BMC Bioinformatics Volume 12 Supplement 12, 2011: Selected articles from the 9th International Workshop on Data Mining in Bioinformatics (BIOKDD). The full contents of the supplement are available online at <http://www.biomedcentral.com/bmcbioinformatics/12?issue=S12>.

#### Author details

<sup>1</sup>Information Systems Department, New Jersey Institute of Technology, University Heights, Newark, New Jersey, USA. <sup>2</sup>Department of Computer Science & Engineering, POSTECH, Pohang, South Korea. <sup>3</sup>School of IT Engineering, Kyungpook National University, Daegu, South Korea.

#### Authors' contributions

MS developed the fundamental idea of the work, performed experiments, validated the results, and wrote the manuscript. HY and WSH participated in the design, the revision of the study and the experiments.

#### Competing interests

The author(s) declare that they have no competing interests.

Published: 24 November 2011

#### References

- Bunescu R, Ge R, Kate JR, Marcotte ME, Mooney RJ, Ramani KA, Wong WY: Comparative experiments on learning information extractors for proteins and their interactions. *Artificial Intelligence in Medicine* 2005, **33**(2):139-155.
- Yang Z, Lin H, Li Y: BioPPISVMEExtractor: A protein-protein interaction extractor for biomedical literature using SVM and rich feature sets. *J Biomed Inform* 2009, **43**:88-96, doi:10.1016/j.jbi.2009.08.013.
- Bennett KP, Demiriz A: Semi-supervised support vector machines. *Proceedings of the 1998 conference on Advances in neural information processing systems II* 1998, 368-374.
- Schohn G, Cohn D: Less is more: Active learning with support vector machines. *ICML '00: Proceedings of the Seventeenth International Conference on Machine Learning* 2000, 839-846.
- Tur G, Dilek Hakkani-Tur D, Schapire RE: Combining active and semi-supervised learning for spoken language understanding. *Speech Communication* 2005, **45**:171-186.
- Ono T, Hishigaki H, Tanigami A, Takagi T: Automated extraction of information on protein-protein interactions from the biological literature. *Bioinformatics* 2001, **17**(2):155-161.
- Huang M, Zhu X, Hao Y, Payan DG, Qu K, Li M: Discovering patterns to extract protein-protein interactions from full texts. *Bioinformatics* 2004, **20**:3604-12.
- Blaschke C, Andrade MA, Ouzounis C, Valencia A: Automatic extraction of biological information from scientific text: protein-protein interactions. *Proc Int Conf Intell Syst Mol Biol* 1999, 60-67.
- Pustejovsky J, Castaño J, Zhang J, Kotecki M, Cochran B: Robust relational parsing over biomedical literature: extracting inhibit relations. *Pac Symp Biocomput* 2002, 362-73.
- Jenssen TK, Laegreid A, Komorowski J, Hovig E: A literature network of human genes for high-throughput analysis of gene expression. *Nature Genetics* 2001, **28**(1):21-8.
- Friedman C, Kra P, Yu H, Krauthammer M, Rzhetsky A: GENIES: a natural-language processing system for the extraction of molecular pathways from journal articles. *Bioinformatics* 2001, **17**(Suppl 1):S74-82.
- Zhou D, He Y, Kwok C: Semi-supervised learning of the hidden vector state model for extracting protein-protein interactions. *Artificial Intelligence in Medicine* 2007, **41**:209-222.
- Song M, Song IY, Hu X, Allen RB: Integrating text chunking with mixture hidden Markov models for effective biomedical information extraction. *International Workshop on Bioinformatics Research and Applications* 2005.
- Temkin JM, Gilder MR: Extraction of protein interaction information from unstructured text using a context-free grammar. *Bioinformatics* 2003, **19**:2046-2053.
- Yakushiji A, Miyao Y, Tateisi Y, Tsujii J: Biomedical information extraction with predicate argument structure patterns. *Proceedings of the Eleventh Annual Meeting of the Association for Natural Language Processing* 2005, 93-96.
- Mitsumori T, Murata M, Fukuda Y, Doi K, Doi H: Extracting protein-protein interaction information from biomedical text with svm. *IEICE Transactions on Information and Systems* 2006, **E89-D**(8):2464-2466.
- Miyao Y, Sagae K, Sætre R, Matsuzaki T, Tsujii J: Evaluating contributions of natural language parsers to protein-protein interaction extraction. *Bioinformatics* 2009, **25**(3):394-400.
- Cui B, Lin H, Yang Z: Uncertainty sampling-based active learning for protein-protein interaction extraction from biomedical literature. *Expert Systems with Applications* 2009, **36**:10344-10350.
- Luo T, Kramer K, Goldgof DB, Hall LO, Samson S, Remsen A, Hopkins T: Active learning to recognize multiple types of plankton. *Journal of Machine Learning Research* 2005, **6**:589-613.
- Burges CJC: A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery* 1998, **2**(2):121-167.
- Joachims T: Transductive inference for text classification using support vector machines. *Proceedings of ICML-99* 1999, 200-209.
- Zhu X: Semi-supervised learning literature survey. *Technical Report 1530 Computer Sciences, University of Wisconsin-Madison*; 2005 [[http://www.cs.wisc.edu/~jerryzhu/pub/ssl\\_survey.pdf](http://www.cs.wisc.edu/~jerryzhu/pub/ssl_survey.pdf)].
- Chapelle O, Sindhvani V, Keerthi SS: Optimization techniques for semi-supervised support vector machines. *Journal of Machine Learning Research* 2008, **9**:203-233.
- Chapelle O, Zien A: Semi-supervised classification by low density separation. *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics* 2005, 57-64.
- Chapelle O, Sindhvani V, Keerthi SS: Branch and bound for semi-supervised support vector machines. *Advances in Neural Information Processing Systems 19* 2007.
- Sindhvani V, Keerthi SS, Chapelle O: Deterministic Annealing for Semi-supervised Kernel Machines. *International Conference on Machine Learning (ICML)* 2006.
- Pyysalo S, Ginter F, Pahikkala T, Koivula J, Boberg J, Jrvinen J, Salakoski T: Analysis of link grammar on biomedical dependency corpus targeted at protein-protein interactions. *Proceedings of the International Workshop on Natural Language Processing in Biomedicine and its Applications* 2004.
- Chapman WW, Bridewell W, Hanbury P, Cooper GF, Buchanan BG: A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of Biomedical Informatics* 2001, **34**:301-310.
- Smith L: Overview of BioCreative II gene mention recognition. *Genome Biology* 2008, **9**(Suppl 2):S2.
- Jenkins MC, Smith D: Conservative stemming for search and indexing. 2005 [<http://fizz.cmp.uea.ac.uk/Research/stemmer/stemmer25feb.pdf>].
- McCallum AC: MALLET: A Machine Learning for Language Toolkit. 2002 [<http://mallet.cs.umass.edu>].
- Lafferty J, Sleator D, Temperley D: Grammatical trigrams: a probabilistic model of link grammar. *Proceedings of the AAAI Conference on Probabilistic Approaches to Natural Language* 1992.
- Kim S, Shin SY, Lee IH, Kim SJ, Sriram R, Zhang BT: PIE: an online prediction system for protein-protein interactions from text. *Nucleic Acids Res* 2008, **36**(Web Server issue):W411-5.

34. Pyysalo S, Ginter F, Heimonen J, Björne J, Boberg J, Järvinen J, Salakoski T: **BiolInfer: A corpus for information extraction in the biomedical domain.** *BMC Bioinformatics* 2007, **8**:50.
35. Fung F, Mangasarian O: **Semi-supervised support vector machines for unlabeled data classification.** *Optimization Methods and Software* 2001, **15**:29-44.
36. Erkan G, Ozgur A, Radev DR: **Semi-supervised classification for extracting protein interaction sentences using dependency parsing.** In *Proceedings of the Conference of Empirical Methods in Natural Language Processing (EMNLP '07)* 2007.

doi:10.1186/1471-2105-12-S12-S4

**Cite this article as:** Song *et al.*: Combining active learning and semi-supervised learning techniques to extract protein interaction sentences. *BMC Bioinformatics* 2011 **12**(Suppl 12):S4.

**Submit your next manuscript to BioMed Central  
and take full advantage of:**

- Convenient online submission
- Thorough peer review
- No space constraints or color figure charges
- Immediate publication on acceptance
- Inclusion in PubMed, CAS, Scopus and Google Scholar
- Research which is freely available for redistribution

Submit your manuscript at  
[www.biomedcentral.com/submit](http://www.biomedcentral.com/submit)

