BIOINFORMATICS ORIGINAL PAPER Vol. 22 no. 17 2006, pages 2136–2142
doi:10.1093/bioinformatics/btl350

Data and text mining

Substring selection for biomedical document classification

Bo Han[1], Zoran Obradovic[1], Zhang-Zhi Hu[2], Cathy H. Wu[2] and Slobodan Vucetic[1,*]

[1]Center for Information Science and Technology, Temple University, Philadelphia, PA 19122, USA and [2]Department of Biochemistry and Molecular & Cellular Biology, Georgetown University Medical Center, Washington DC 20007, USA

Received on March 2, 2006; revised on June 4, 2006; accepted on June 23, 2006
Advance Access publication July 12, 2006
Associate Editor: Thomas Lengauer

ABSTRACT

Motivation: Attribute selection is a critical step in development of document classification systems. As a standard practice, words are stemmed and the most informative ones are used as attributes in classification. Owing to high complexity of biomedical terminology, general-purpose stemming algorithms are often conservative and could also remove informative stems. This can lead to accuracy reduction, especially when the number of labeled documents is small. To address this issue, we propose an algorithm that omits stemming and, instead, uses the most discriminative substrings as attributes.

Results: The approach was tested on five annotated sets of abstracts from iProLINK that report on the experimental evidence about five types of protein post-translational modifications. The experiments showed that Naive Bayes and support vector machine classifiers perform consistently better [with area under the ROC curve (AUC) accuracy in range 0.92–0.97] when using the proposed attribute selection than when using attributes obtained by the Porter stemmer algorithm (AUC in 0.86–0.93 range). The proposed approach is particularly useful when labeled datasets are small.

Contact: vucetic@ist.temple.edu

Supplementary Information: The supplementary data are available from www.ist.temple.edu/PIRsupplement

1 INTRODUCTION

Information retrieval from biomedical documents is a challenging problem owing to the richness and complexity of biomedical terminology and the large volume of published work in the discipline. The specific application that motivated this study is related to the functional annotation of biological molecules. PIR-PSD (Protein Sequence Database created by Protein Information Resource) is a comprehensive and expertly curated database of classified and functionally annotated protein sequences (Wu *et al*., 2003), which is now being merged into the UniProt (Wu *et al*., 2006). For each PSD protein entry, functional annotations including the active sites, binding sites and modification sites are provided together with annotation confidence tags such as 'experimental' or 'predicted'. Recently, evidence attributions to the literature confirming those 'experimental' annotations have been conducted by manually surveying literature citations in the entry from which the annotations were originally derived (Hu *et al*., 2004). This property of attributing experimental annotations to literature evidence greatly enhances the quality and value of the resource. However, manual annotation is a tedious and time-consuming process as it requires curators to review large numbers of candidate papers. As the volume of sequence data and scientific literature continues to grow exponentially, this manual process hampers the ability to keep the annotations up-to-date (Hu *et al*., 2004).

A very attractive option for expediting literature surveys is the use of advanced text mining and information retrieval tools aimed at rapid and accurate pre-screening of documents. There is active research in the area of biomedical text mining aimed at facilitating information retrieval from biological literature. At 2002 KDD Cup, given 862 journal articles curated by FlyBase, contest teams were expected to determine whether 213 test papers contained experimental evidence about gene products. The winning and honorable mention teams all manually or semi-automatically constructed rules or patterns and used them to evaluate the test documents (Regev *et al*., 2003; Shi *et al*., 2003; Ghanem *et al*., 2003). Similarly, PIR recently built a rule-based literature mining system RLIMS-P for finding phosphorylation information from MEDLINE abstracts (Hu *et al*., 2005). This approach required the involvement of domain experts in identifying informative templates for identification of the specific textual objects. While successful for the specific applications for which it is tuned, the rule-based approach cannot easily be extended to an efficient and scalable working model on other biomedical datasets.

Fully automatic and scalable text classification algorithms provide an alternative to the rule-based approaches. Wilbur (2000) used 3121 positive documents and 117 476 negative documents to build a classification model to rank MEDLINE abstracts based on their similarities with the information in the restriction enzyme database REBASE. More than 10 000 attributes for classification were selected from individual words, adjacent pairs of words and MeSH terms. Marcotte *et al*. (2001) used 260 positives and 65 807 negatives to build a ranking model that used 500 selected words as attributes. Dobrokhotov *et al*. (2003) built a probabilistic classifier based on 2188 abstracts from 3 categories (15% in category 'relevant', 70% in category 'irrelevant' and another 15% in category 'unclear'). The discriminating attributes were selected from words processed by the XEROX natural language processing (NLP) tool. Aphinyanaphongs and Aliferis (2005) developed a support vector machine (SVM) classifier that used stemmed words as attributes from 396 positive and 15 407 negative papers. The common property of these text classification applications is that the number of labeled documents

*To whom correspondence should be addressed.

was moderate to high, and that the extracted attributes were based on word occurrences. Most often, the documents were preprocessed by word stemming in order to reduce the number of unique words without significant loss of information.

When labeled datasets are small (i.e. containing up to a few hundred labeled documents), general-purpose stemming algorithms can result in a dictionary that contains many infrequently occurring but informative words. Subsequent attribute selection would tend to remove those infrequent informative words and the resulting classifier would have reduced accuracy. The vast and complex biomedical terminology only exacerbates this problem. Specifically, semantically related biomedical terms that share the same stem or morpheme are often not reducible to the same stem using popular stemming algorithms. As an example, 'phosphorylated', 'phosphate', 'phosphorylatable', 'phosphorylase', 'phosphopeptides' all contain the same stem 'phosph', which can be highly useful for identification of documents related to protein phosphorylation. However, Porter stemmer (Porter, 1980), one of the most popular stemming algorithms, can only reduce words from the example to 'phosphoryl', 'phosphat', 'phosphorylat', 'phosphorylas', 'phosphopeptid', respectively. As a result, five attributes would be constructed to represent occurrences of these words. If the labeled dataset is small, it is likely that none of these attributes are selected for classification owing to their infrequent occurrence.

In addition to the difficulties stemming algorithms have in producing maximally informative stems, their inherent drawback is the removal of suffixes based on general grammar rules. Biomedical terms often contain informative suffixes. For example, 'ase' is a suffix common to proteins that function as enzymes (e.g. kinase) and its occurrence is necessary in the identification of documents that are related to such enzymes. This information is not recoverable if popular stemming algorithms are used. The drawbacks of stemming algorithms have been observed by other researchers. Nenadic *et al*. (2003) studied the performance of a text classification approach by comparing different types of attribute extraction procedures. Their results show that the use of standard stemming algorithms as a preprocessing step does not improve accuracy of biomedical text classification.

It is evident that new and unique methods need to be developed for the extraction of informative attributes from biomedical documents. Andrade *et al*. (1998) proposed a new stemming algorithm for biomedical documents by applying a set of simple rules, such as two words with length larger than five have the same stem if they have the same prefix and their suffixes differ in at most two characters. Nenadic *et al*. (2003) and Rice *et al*. (2005) applied an enhanced version of the C-value method to extract domain-specific terms. The method needs rules of conflation to reduce term variants; it yields good performance only if sufficient training data are available.

In this paper, we propose an effective and simple attribute selection algorithm that derives attributes from substrings. The algorithm explores all substrings in labeled documents and selects the ones that most successfully discriminate between positive and negative documents. Attributes are derived as counts of the most informative non-redundant substrings. A classification model (i.e. the Naive Bayes classifier) is constructed from the labeled dataset using the selected attributes. The appeal of this procedure is that attribute selection is driven directly by the labeled dataset. In addition, substring-based attribute extraction provides increased flexibility in comparison with the traditional word-based procedures.

iProLINK (Hu *et al*., 2004) is a new PIR resource that provides multiple annotated literature corpora to facilitate text mining research in the areas of literature-based database curation, named entity recognition, and protein ontology development. We evaluated the approach on five iProLINK datasets of tagged abstracts corresponding to post-translational protein modifications (PTMs), i.e. acetylation, glycosylation, methylation, phosphorylation and hydroxylation. The presented experimental evidence suggests that classifiers that use the proposed attribute extraction algorithm perform better than the ones that use attributes obtained by the standard Porter stemmer algorithm.

2 METHODS

The proposed substring selection algorithm is based on enumerating the usefulness of each frequently occurring substring and selecting the most informative subset of substrings as attributes for text classification.

2.1 Attribute extraction

Given a corpus of N documents $\{d_i\ ,\ i = 1,\ldots,N\}$, each document d_i is parsed into a set of words w_{ij}, $i = 1,\ldots,N\ ,\ j = 1,\ldots,K_i$, that are separated by spacer symbols. K_i is number of words in document d_i. We considered two approaches for attribute extraction: (1) Word-Based, which is the standard approach based on Porter stemming and (2) Substring-Based, which is the proposed approach.

Porter stemming is a highly effective, simple algorithm that removes word suffixes in order to reduce related words (e.g. connected, connection) to the same stem (e.g. connect). The Word-Based algorithm begins with the construction of a dictionary of unique stemmed words from the corpus. Each document d_i is then represented as a vector $f_i = [f_{i1},\ldots,f_{iK}]$, where f_{ik} is the count of stem s_k in d_i, and K is the dictionary size. The Porter stemmer is appropriate for general-subject English documents because it manages to significantly reduce the dictionary size without the excessive conflation of unrelated words. Owing to complex terminology and unconventional naming, standard grammar-based stemmers (e.g. Porter stemming) can be too conservative and result in a variety of closely related but rarely occurring terms. In addition, it can result in the removal of potentially useful suffixes.

In order to avoid the pitfalls of grammar-based stemming and to achieve a highly flexible attribute extraction, the Substring-Based approach is proposed. It constructs a dictionary of unique substrings which occur in the document corpus. A substring is defined as a consecutive list of symbols within a word. As in the Word-Based approach, given a dictionary, each document d_i is represented as vector $f_i = [f_{i1},\ldots,f_{iK}]$, where f_{ik} is the count of substring s_k in d_i, and K is the dictionary size.

Efficient procedures for construction of word or substring dictionary are available. For example, suffix trees can be used where each node represents a unique substring s, i.e. a suffix of its parent node. The suffix tree construction takes time linear to a number of words in the corpus. A node representing substring s_k is assigned a frequency vector $f_k = [f_{1k},\ldots,f_{Nk}]$, where f_{ik} is the count of substring s_k in document d_i and N is number of documents in the corpus.

Since each word w contributes $|w|(|w|+1)/2$ substrings, it is evident that the Substring-Based dictionary is larger than the Word-Based dictionary. For example, for the five datasets (see Section 3.1) used in the experiments the total number of substrings is on average 39 times larger than the number of words and the Substring-Based dictionary size is on average about 15 times larger than the Word-Based dictionary size. Additionally, to find word w in a dictionary requires $|w|$ comparisons, while to find all of its substrings takes $|w|(|w|+1)/2$ comparisons. For example, since the average word length in the five datasets from Section 3.1 is 7.2 and 6.2 before and after Porter stemming, the Substring-Based approach is $\sim$4 times slower than the Word-Based approach. However, since the Substring-Based approach is aimed towards problems with small sets of labeled documents,

the added computational complexity should not be considered as a serious drawback.

2.2 Attribute selection

By observing that rare substrings could not be used successfully to improve classification performance, all substrings that occur in less than three documents are excluded from further consideration. In the following step, the relevance of each frequent substring is evaluated. In principle, any of the numerous feature selection algorithms from machine learning could be suitable for this step.

Information gain (IG) was used as the benchmark because it is considered very appropriate for text classification (Yang and Pederson, 1997). Given a multi-valued attribute $X \in \{x_1, \ldots, x_K\}$ and class variable $Y \in \{1, \ldots, C\}$, the IG of attribute X is calculated as

$$\mathrm{IG}(X) = \sum_{i=1}^{K} \sum_{j=1}^{C} p(X = x_i, Y = j) \cdot \log_2 \frac{p(X = x_i, Y = j)}{p(X = x_i) \cdot p(Y = j)}.$$

The attributes with high IG value are considered relevant.

The IG criterion is biased towards frequently occurring attributes. To address this drawback, we used the Wilcoxon rank-sum test (WRST) to measure substring relevance. The relevance of substring s_k is obtained by sorting the documents by their attribute values, assigning them ranks and summing up the ranks of the positive and negative documents. If the attribute is not relevant, the average ranks of positive and negative documents should be similar. Using the Gaussian approximation, the P-value of the test is computed using complementary error function (Cody, 1969) erfc($|Z|$), where $Z = (U - \mathrm{E}(U))/\mathrm{sqrt}(\mathrm{Var}(U))$, U is sum of ranks of positive samples, and E(U) and Var(U) are the mean and the variance of U. A small p-value indicates high attribute relevance.

2.3 Removing redundant attributes

Attributes obtained by the substring-based approach and selected by either IG or WRST are likely to have high degree of redundancy. The redundancy should be reduced because it can have an adverse effect on learning. Possible redundancy-reduction procedures include latent semantic analysis (Berry *et al*., 1995) or heuristics that ensure that no two substrings in the selected set are prefix, suffix or subset of each other. Based on our pilot study, we decided to apply a rather simple but effective approach that uses a correlation measure. Given a threshold T, the approach ensures that no two attributes have a correlation coefficient that exceeds T. The procedure compares each pair of substring attributes, and if their correlation is above T, an attribute with smaller relevance is removed.

2.4 Ranking algorithms

For document ranking, we considered SVMs and Naive Bayes classifiers. These classification algorithms are directly applicable to ranking because their outputs are correlated with the posterior probability of a positive document. SVMs (Vapnik, 1995) are optimized to find the decision hyperplane with the maximum separation margin between positive and negative data points. By representing document d_i with pair (f_i, y_i), where $f_i = [f_{1i} \ldots, f_{Ki}]$ is attribute vector, f_{ki} is count of substring s_k in document d_i, $y_i = -1$ for negative documents and $y_i = +1$ for positive documents, the output of SVM for a document with attribute vector $\boldsymbol{f} = [f_1, \ldots, f_K]$ is calculated as

$$\mathrm{SVM}(f) = b + \sum_{i=1}^{N_S} \alpha_i y_i K(f_i, f),$$

where N_S is number of support vectors selected from training data, α_i, $i = 1, \ldots, N_S$, and b are model parameters obtained by optimization, and K is an appropriate kernel function. The reported experimental evidence suggests that SVMs are very successful in the classification of high-dimensional data, and that SVMs with linear kernels are often as accurate as SVM with non-linear kernels in text classification (Joachims, 1998).

Naive Bayes is a simple classifier that operates under the assumption that attributes are conditionally independent given a class variable. In this study we considered a multinomial Naive Bayes classifier, known to be appropriate for text classification (McCallum and Nigam, 1998). For binary classification, the output of multinomial Naive Bayes classifier for a document with the attribute vector $\boldsymbol{f} = [f_1, \ldots, f_K]$ can be expressed as

$$\mathrm{NB}(f) = \log \frac{P(y = +1 \,|\, f)}{P(y = -1 \,|\, f)} = \log \frac{P^+}{P^-} + \sum_{k=1}^{K} f_k \cdot \log \frac{P_k^+}{P_k^-}$$

where P^+ and P^- are fractions of positive and negative training documents, and P_k^+ and P_k^- are frequencies of substring s_k in positive and negative documents obtained as

$$P_k^+ = \frac{1 + \sum_{y_i = +1} f_{ki}}{K + \sum_{k=1}^{K} \sum_{y_i = +1} f_{ki}}, P_k^- = \frac{1 + \sum_{y_i = -1} f_{ki}}{K + \sum_{k=1}^{K} \sum_{y_i = -1} f_{ki}}.$$

The top ranked documents are the ones with highest SVM(f) or NB(f). It is worth noting that, for the purposes of documents ranking, the priors P^+ and P^- can be neglected. This is a highly useful property because the fractions of positive and negative training documents are unlikely to be related to their fractions found in unlabeled documents.

2.5 Performance measures

The main objective of document ranking is to achieve high ranking of positive documents. To evaluate ranking quality we used three measures that are based on ROC curves. An ROC curve measures the trade-off between true positive (TP; fraction of positives predicted as positives) and false positive (FP; fraction of negatives predicted as positives) prediction rates for different prediction cutoffs. Given the prediction cutoff θ, all documents with prediction [e.g. *SVM*(f), *NB*(f)] above θ are considered positive and all below negative. If θ is very small, no positives are predicted, and TP = 0 and FP = 0, while if θ is very large TP = 1 and FP = 1.

Area under the ROC curve (AUC). Predictors that achieve high TP over a range of FP are considered accurate—AUC measures exactly this aspect of prediction quality. Perfect predictors achieve AUC = 1, while predictors that provide random predictions have AUC = 0.5.

*TOP*10. It is calculated as the number of TP among the 10 highest ranked documents. This is a very relevant measure of ranking quality since high TOP10 accuracy is likely to motivate a user to continue using the system.

*FP*80. It is calculated as FP rate when the predictor achieves 80% TP rate. This value is relevant for extensive uses of the ranking system where the goal is to retrieve majority of positive documents with minimal effort.

3 RESULTS

3.1 Datasets

Five sets of documents summarized in Table 1 were used to evaluate the proposed procedure. They consisted of MEDLINE abstracts related to acetylation, glycosylation, methylation, phosphorylation and hydroxylation PTMs. Each abstract was labeled by PIR curators either as positive, if it reported experimental evidence about the PTM, or as negative, if it did not. The sizes of the datasets were moderate, ranging from 160 to 923, and were characterized by significant class imbalance, ranging from 5:1 to 17:1. To further evaluate the classification performance, we have used 1088 untagged abstracts obtained from 347 PIR-PSD entries for proteins that contained glycosylation sites.

All documents were preprocessed in the following way: upper case letters were mapped to lower case letters; all digits were mapped to digit symbol D; and all special symbols excluding

Table 1. Summary of the five PTM datasets

PTM types	Positives	Negatives
Acetylation	55	868
Glycosylation	41	711
Hydroxylation	27	133
Methylation	27	171
Phosphorylation	79	389

'–' were mapped to spacer symbol B. A total of 524 common words (e.g. 'and', 'of', etc.) were removed from the documents.

3.2 Experimental setup

For each of the five datasets from Table 1, we performed eight groups of experiments on each combination from (Word-Based, Substring-Based attribute extraction) × (SVM, Naive Bayes classifier) × (IG, WRST attribute selection). For each choice, we estimated AUC, TOP10 and FP80 accuracies using cross-validation. The procedure consisted of splitting the labeled documents randomly into $K = 5$ equal subsets. One of the subsets was used for accuracy testing while the remaining ones were used for training. The process was repeated $K = 5$ times, each time using a different subset for testing. The whole 5-cross-validation procedure was repeated 20 times, each time using different initial split of the labeled abstracts into K subsets. The average accuracy over the 100 experiments was reported as the accuracy estimate.

While application of the Naive Bayes classifiers was straightforward, use of SVMs required appropriate data preprocessing and parameter selection. For our experiments, we used Spider SVM software (http://www.kyb.tuebingen.mpg.de/bs/people/spider/) that runs SVMLight (Joachims, 1999) in the background. Before training, each attribute was scaled to range [0,1]. Since our datasets were highly imbalanced, we explored several values of the balanced_ridge parameter r, which modifies kernel matrices by balancing influence of positive and negative examples. It adds $r{\cdot}N^+/(N^+ + N^-)$ to positive examples at the diagonal of the kernel matrix and $r{\cdot}N^-/(N^+ + N^-)$ to the negative examples, where N^+ is the number of positives, and N^- is the number of negatives. We explored a range of r values between 2^{-8} and 2^8, and determined that $r = 10^{-4}$ works well on all five subsets. The default SVMLight value for the slack variable was used in all experiments.

3.3 Accuracy comparison

In Table 2 we show the comparison between eight different ranking algorithms. The first two (WB-NB-IG and WB-SVM-IG) used Word-Based (WB) attribute extraction and Information Gain (IG) attribute selection. SVM denotes SVMs and NB Naive Bayes ranking algorithm. The second two (WB-NB-WRST and WB-SVM-WRST) used Word-Based attribute extraction and Wilcoxon test (WRST) attribute selection. The third two (SB-NB-IG and SB-SVM-IG) used Substring-Based (SB) attribute extraction and Information Gain attribute selection. The final two (SB-NB-WRST and SB-SVM-WRST) used Substring-Based attribute extraction and Wilcoxon test attribute selection.

For SB algorithms, our experiments revealed that the accuracy was the highest for the correlation threshold $T = 0.99$, that it was stable in the range between 0.8 and 0.99, and that it dropped slightly beyond this range. Therefore, $T = 0.99$ threshold was used in all the experiments of Table 2. Our experiments showed that the optimal IG threshold was 0.02 and it was fixed at this value in all experiments of Table 2. The optimal p-value threshold for WRST attribute selection was 0.15 and it was fixed for all experiments of Table 2. These threshold values resulted in selection of ∼100 word-based and ∼1000 substring-based attributes in all of the experiments.

Table 2. Accuracies of Word-based and Substring-based classifiers (WB, Word-Based; SB, Substring-Based; NB, Naive Bayes; SVM, support vector machines; IG, information gain; WRST, Wilcoxon test)

Method	FP80	AUC	TOP10
(a)Acetylation group			
WB-NB-IG	0.237 ± 0.018	0.855 ± 0.005	3.00 ± 0.38
WB-SVM-IG	0.200 ± 0.031	0.869 ± 0.035	3.20 ± 0.39
WB-NB-WRST	0.236 ± 0.010	0.854 ± 0.013	3.02 ± 0.40
WB-SVM-WRST	0.239 ± 0.028	0.851 ± 0.030	2.98 ± 0.31
SB-NB-IG	0.229 ± 0.009	0.864 ± 0.007	3.08 ± 0.36
SB-SVM-IG	0.168 ± 0.022	0.882 ± 0.014	4.16 ± 0.17
SB-NB-WRST	**0.146 ± 0.010**	**0.916 ± 0.007**	**4.64 ± 0.33**
SB-SVM-WRST	0.171 ± 0.024	0.893 ± 0.013	4.60 ± 0.30
(b) Glycosylation group			
WB-NB-IG	0.152 ± 0.032	0.924 ± 0.006	5.20 ± 0.30
WB-SVM-IG	0.176 ± 0.039	0.883 ± 0.009	4.56 ± 0.37
WB-NB-WRST	0.121 ± 0.026	0.926 ± 0.005	5.36 ± 0.27
WB-SVM-WRST	0.153 ± 0.038	0.890 ± 0.008	4.64 ± 0.29
SB-NB-IG	0.052 ± 0.002	0.953 ± 0.004	5.80 ± 0.24
SB-SVM-IG	0.118 ± 0.055	0.926 ± 0.022	5.40 ± 0.22
SB-NB-WRST	**0.035 ± 0.007**	**0.968 ± 0.004**	**6.28 ± 0.18**
SB-SVM-WRST	0.106 ± 0.076	0.929 ± 0.017	5.40 ± 0.37
(c) Hydroxylation group			
WB-NB-IG	0.158 ± 0.041	0.888 ± 0.012	3.96 ± 0.36
WB-SVM-IG	0.286 ± 0.095	0.803 ± 0.034	3.80 ± 0.54
WB-NB-WRST	0.202 ± 0.032	0.858 ± 0.018	3.86 ± 0.32
WB-SVM-WRST	0.325 ± 0.091	0.795 ± 0.039	3.78 ± 0.46
SB-NB-IG	0.165 ± 0.015	0.911 ± 0.003	4.44 ± 0.14
SB-SVM-IG	0.233 ± 0.031	0.870 ± 0.029	4.40 ± 0.20
SB-NB-WRST	**0.083 ± 0.010**	**0.948 ± 0.004**	**4.76 ± 0.09**
SB-SVM-WRST	0.173 ± 0.042	0.872 ± 0.025	4.50 ± 0.22
(d) Methylation group			
WB-NB-IG	0.244 ± 0.040	0.866 ± 0.015	3.88 ± 0.23
WB-SVM-IG	0.209 ± 0.026	0.854 ± 0.020	3.60 ± 0.36
WB-NB-WRST	0.225 ± 0.028	0.880 ± 0.014	3.94 ± 0.26
WB-SVM-WRST	0.202 ± 0.022	0.864 ± 0.015	3.78 ± 0.38
SB-NB-IG	0.167 ± 0.038	0.916 ± 0.015	4.40 ± 0.19
SB-SVM-IG	0.208 ± 0.040	0.863 ± 0.027	4.20 ± 0.32
SB-NB-WRST	**0.089 ± 0.010**	**0.940 ± 0.005**	**4.60 ± 0.25**
SB-SVM-WRST	0.196 ± 0.041	0.879 ± 0.029	4.40 ± 0.17
(e) Phosphorylation group			
WB-NB-IG	0.220 ± 0.006	0.895 ± 0.008	6.16 ± 0.18
WB-SVM-IG	0.103 ± 0.011	0.896 ± 0.008	6.20 ± 0.20
WB-NB-WRST	0.215 ± 0.009	0.895 ± 0.011	6.20 ± 0.10
WB-SVM-WRST	0.130 ± 0.012	0.898 ± 0.013	6.24 ± 0.22
SB-NB-IG	0.096 ± 0.005	0.917 ± 0.003	6.48 ± 0.26
SB-SVM-IG	0.152 ± 0.040	0.909 ± 0.016	6.56 ± 0.28
SB-NB-WRST	**0.088 ± 0.005**	**0.925 ± 0.002**	**6.80 ± 0.31**
SB-SVM-WRST	0.114 ± 0.025	0.911 ± 0.010	6.65 ± 0.36

Table 3. Comparison of top 15 selected words/substrings

Dataset	Top 15 selected words
(a) Words selected by Wilcoxon test	
Acetylation	amino (+), residu (+), termin (+), acid (+), terminu (+), peptid (+), mass (+), gene (−), acetyl (+), primary (+), phase (+), clone (−), cdna (−), code (−), dna (−)
Glycosylation	carbohydr (+), acid (+), residu (+), determine (+), clone (−), attach (+), peptid (+), chain (+), human (−), region (−), cdna (−), mrna (−), sequence (+), posit (+), biochem (+)
Hydroxylation	residu (+), acid (+), type (−), protein (+), gene (−), domain (−), marin (+), encod (−), clone (−), spectrometri (+), molecul (−), helic (−), dna (−), chemic (+), degrade (+)
Methylation	residu (+), amino (+), protein (+) , modify (+), gene (−), biochem (+), termin (+), clone (−), block (+), reaction (+), genom (−), degrade (+), transcript (−), terminu (+), mrna (−)
Phosphorylation	phosphoryl (+), site (+), sequence (−), serin (+), gene (−), clone (−), residu (+), cdna (−), peptid (+), vitro (+), nucleotide (−), protein (+), code (−), active (+), dna (−)
(b) Substrings selected by the Wilcoxon test (together with the representative words)	
Acetylation	**acety** {acetylated}, **termin** {terminal, determinated}, **ked** {blocked, linked, flanked}, **residue**, **ac-** {ac-val-asp-ser}, **pro-D** {pro-11}, **lock** {blocked}, **omo** {homology, homogeneous, isomorphic}, **blo** {blocked, blot}, **han** {tryptophan, enhance}, **acy** {acyl}, **etermined** {determined}, **acid**, **nal** {terminal, analysis},**pep** {peptide}
Glycosylation	**asn** {asn-45, asn103}, **carboh**, **asp** {asparagines, asp-85}, **gos** {oligosaccharide}, **glycos**, **omor** {myomorph, hystricomorph}, **rate** {carbohydrate, demonstrate}, **ched** {attached, enriched}, **ache** {attached, reaches}, **ragi** {asparagine}, **opep** {endopeptidase}, **chari** {oligosaccharide}, **glycope** {glycopeptidase}, **hydra** {carbohydrate}, **syl** {lysyl}
Hydroxylation	**hydroxyl**, **position**, **xyp** {hydroxyproline}, **xya** {hydroxyasparagine}, **rome** {spectrometry}, **peptide**, **fas** {fasciola}, **mica** {chemical}, **residue**, **sole** {isoleucine}, **cyc** {cycle}, **mollu** {mollusc}, **geneo** {heterogeneous}, **hro-** {erythro-beta}, **trom** {spectrometry}
Methylation	**methyl**, **residue**, **parti** {partial, aspartic}, **modif**, **geneo** {heterogeneous}, **lysine** {trimethyllysine}, **ified** {modified, identified}, **plee** {spleen}, **post** { posttranslational}, **termin** {terminus}, **sine** (lysine, tyrosine}, **hem** {chemical}, **ttr** {posttranslational, attracants}, **bac** {bacterial}, **lock** {blocked}
Phosphorylation	**phosph**, **sit** {site, position}, **serin**, **amp-** {camp-dependent}, **lys** {lysine}, **kinase**, **oser** {phosphoserine}, **residue**, **opep** {phosphopeptide}, **ser-** {gln-ser-gly, ser-38}, **threo** {threonine}, **rad** {radioactive, degradation}, **dig** {digestion}, **vitro**, **ivo** {vivo}

Regardless of the accuracy measure, SB-NB-WRST consistently outperformed other alternatives. Naive Bayes was more successful than SVM on all datasets. On methylation and hydroxylation data the difference was quite substantial. It seems that the main problem with SVM was high class imbalance—the difference between SVM and NB was the smallest for the phosphorylation data.

SB-NB-IG was superior to WB-NB-IG which indicates that the proposed SB attribute selection is useful. The difference between SB and WB approaches was largest on methylation and hydroxylation which were the smallest sets. This result confirms that substring attributes perform better than word stems on small datasets. WRST resulted in higher accuracy than the IG attribute selection. The difference was the highest on acetylation and glycosylation datasets that were the most unbalanced.

While AUC accuracy was relatively stable over all experiments, FP80 measure showed large variability ranging from 4 to 15%. The reason for this was unstable behavior of the initial portions (small FP rates) of ROC curves. TOP10 accuracy of the most accurate SB-NB-WRST ranged from 4.6 to 6.8. This is a very promising result considering the high class imbalance in each of the five datasets.

3.4 Comparison of attribute selection

In Table 3 (Words selected by Wilcoxon test), we list the top 15 selected words for each of the five PTMs based on the Wilcoxon test. We distinguish positive and negative attributes as the ones that are more frequent in positive and negative documents, respectively. For comparison, in Table 3 (Substrings selected by the Wilcoxon test) we list the top 15 substrings with the Substring-Based approach that uses WRST criterion.

As seen, the top substrings are most often directly related to corresponding post-translational modification types. For example, the substrings include stems of PTM names, such as ‘acety’, ‘glycos’, ‘hydroxyl’, ‘methyl’ and ‘phosph’. In addition, all of the top 15 substrings in every dataset were positive attributes, which is in contrast with the Word-Based method that often selects negative attributes. In most cases, the negative words were gene-related (e.g. ‘gene’, ‘clone’, ‘cdna’, ‘transcript’) and their presence was indicative of documents that do not address PTMs. While negative attributes are useful, we think that positive attributes allow more focused detection of documents focused to PTMs among a diverse set of documents.

Table 3 (Words selected by Wilcoxon test) also illustrates that Porter stemmer can be conservative—among the top 15 words in the Acetylation and Methylation datasets are both ‘termin’ (stemmed from ‘terminal’) and ‘terminu’ (stemmed from ‘terminus’). From Table 3 (Substrings selected by the Wilcoxon test) it can be seen that the Substring-Based approach extracted only substring ‘termin’ that is sufficient to represent both ‘terminal’ and ‘terminus’ words.

An interesting property of the Substring-Based method is the occasional selection of substrings that are parts of the same word (e.g. ‘asp’ and ‘ragi’ are both part of ‘asparagine’). While this might seem inappropriate, our results showed that presence of such related substrings did not adversely influence the accuracy. Besides substrings that are directly stemmed from the PTM types,

other substrings such as 'residue' are often related to protein sequence properties common to all PTMs, which can discriminate documents characterizing protein sequences from other types of biomedical documents. Interestingly, some amino acid residues specific for each PTM are in the list, e.g. 'ragi' (for asparagine in glycosylation), 'sole' (isoleucine for hydroxylation), 'lysine' (for methylation), 'serin' and 'lys' (serine and lysine for phosphorylation).

We further illustrate the usefulness of the Substring-Based approach by considering a specific example from iProLINK (http://pir.georgetown.edu/iprolink/). The curators labeled abstract PMID 2606104 in the Acetylation group as positive. They also marked the evidence tag (showed in italics) within the following passage of the abstract: 'The primary structure of glucose-6-phosphate dehydrogenase from rat liver has been determined, showing the mature polypeptide to consist of 513 amino acid residues, with an *acyl-blocked N-terminus*'. Interestingly, although the evidence tags were not used in learning, 3 of the top 15 acetylation substrings in Table 3 (Substrings selected by the Wilcoxon test), 'acy' (ranked at 13), 'lock' and 'termin', occur within the evidence tag. This is a strong indication that, in addition to document ranking, the Substring-Based algorithm could also be helpful in evidence tagging. For example, curators could be aided by highlighting portions of the text with high concentration of highly ranked substrings.

3.5 Ranking of unlabeled abstracts

Using the SB-NB-WRST predictor for ranking of glycolysation documents the ranking system was applied to the 1088 untagged abstracts listed in the 347 PIR-PSD protein entries with glycosylation sites.

The top and bottom 50 ranked abstracts were selected, merged and shuffled into the list of 100 abstracts (Supplementary Table unlabeled glycosylation abstracts). PIR curators were then asked to read and label these 100 abstracts without knowing the actual ranking by our system. The results showed 43 out of the highest ranked 50 abstracts were labeled as positive by the PIR curators, while all of the lowest ranked 50 abstracts were labeled as negative. Further analysis showed that abstracts of the seven false positive papers mostly describe characterization of protein primary sequences and secondary structures, which were the common themes of the positive training abstracts. For example, the top-ranked negative abstract (ranked as 22nd) contain information about protein sequences and contain informative substrings, such as 'terminal' and 'residue'. These results are very encouraging and indicate that our ranking system is successful in reducing the cost of curation.

From the 43 positive abstracts, we also observed that evidence related to protein glycosylation is being described in various ways. Table 4 shows a list of the glycosylation-related words that were manually extracted from the 43 abstracts, including the number of abstracts (out of 43) in which they occurred. For example, 19 of the 43 abstracts used only one of the keywords from the table, 3 of which used only the less-common words such as 'glycan moiety' and 'disaccharide' when referring to glycosylation. Such a variety of words, coupled with their internal similarity (most of them contain substring 'glyc'), clearly speaks in favor of the Substring-Based approach.

In comparison, we used Word-Based approach WB-NB-WRST to rank the same untagged abstracts. The results show that 34 out of the highest ranked 50 abstracts are true positives, while there was 1 false negative abstract in the lowest ranked 50 abstracts. This is a further confirmation that Substring-Based approach performs better than Word-Based approach in our application.

Table 4. Glycosylation-related keywords from the highest-ranked positive abstracts

Keyword	No. of abstracts
Carbohydrate	19
Glycosylate	16
Oligosaccharide	13
Glycopeptide	10
Glycoprotein	7
Glycosylation	5
Glycan Moiety, disaccharide	3

Currently, biomedical researchers and curators most often rely on keyword search (e.g. MEDLINE, Google) to retrieve relevant information. As evident from Table 4, it is often difficult to compose an appropriate keyword-based query, and approaches such as the proposed one offer a promising alternative for efficient biomedical literature search.

4 DISCUSSION

Classification and ranking of biomedical documents is challenging owing to the depth and complexity of biomedical terminology. We have observed that traditional Word-Based stemming algorithms, such as the Porter stemmer, have several drawbacks when used as a preprocessing step in classification of biomedical documents. The two main issues that arise are that the conflation of biomedical terms by stemming is too conservative and that stemming might result in the removal of informative suffixes. The aim of this paper is to show that stemming can be successfully replaced by a procedure that automatically selects the most informative substrings from a set of labeled documents.

The proposed classification/ranking system was evaluated on five PIR annotated datasets. These datasets are representative of a class of biomedical text mining problems. They are related to tasks of information extraction from large text collections (e.g. MEDLINE) in which it is difficult to express the search goals in terms of keyword-based queries. In this case, it is probable that a user would attempt various queries and obtain long lists of retrieved documents. The user would then start reading the retrieved documents and, through the process, label a number of them as relevant or irrelevant. Documents labeled in this way open up an opportunity to apply text classification systems that rearrange the unread documents in order of their relevance. Using our ranking system, most relevant documents would appear near the top of the list and significantly reduce human effort in literature survey.

Our experiments show that the proposed substring-based approach is highly effective even when relatively small labeled datasets are available for learning. This result is in contrast with the behavior of traditional Word-Based algorithms that require large sets of labeled documents in order to approach the accuracy of the Substring-Based algorithm.

It is worth comparing the current ranking system with the RLIMS-P text mining tool (Hu *et al*., 2005) that achieved high

recall (96%) and precision (88%) in classification of phosphorylation documents. Our system was less accurate on the same data, with recall of 80 and precision of 92% [SB-NB-WRST row in Table 2 (Phosphorylation group)]. However, RLIMS-P is a rule-based system that relies on manual selection of discriminative rules for information extraction. This is acceptable for specialized applications such as the large-scale text mining project at PIR for extracting specific annotations of protein post-translational modifications such as phosphorylation from MEDLINE abstracts. However, the proposed Substring-Based system is fully automatic, and so is much easier for application over a wider range of biomedical text mining tasks.

ACKNOWLEDGEMENTS

This project is funded, in part, under a grant with the Pennsylvania Department of Health. The Department specifically disclaims responsibility for any analyses, interpretations or conclusions. The curated literature mining resource for this study was supported in part by grant U01-HG02712 from the National Institutes of Health. Funding to pay the Open Access publication charges was provided by a grant with the Pennsylvania Department of Health.

Conflict of Interest: none declared.

REFERENCES

Andrade,M.A. *et al*. (1998) Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families. *Bioinformatics*, **14**, 600–607.

Aphinyanaphongs,Y. and Aliferis,C.F. (2005) Text categorization models for retrieval of high quality articles in internal medicine. *J. Am. Med. Inform. Assoc*., **12**, 207–216.

Berry,M.W. *et al*. (1995) Using linear algebra for intelligent information retrieval. *SIAM Rev*., **37**, 573–595.

Cody,W.J. (1969) Rational Chebyshev approximations for the error function. *Math. Comp*., **22**, 631–638.

Dobrokhotov,P.B. *et al*. (2003) Combining NLP and probabilistic categorization for document and term selection for Swiss-Prot medical annotation. *Bioinformatics*, **19** (Suppl. 1), i91–i94.

Ghanem,M.M. *et al*. (2003) Automatic scientific text classification using local patterns: KDD Cup 2002 (task 1). *SIGKDD Explor. Newslett*., **4**, 95–96.

Hu,Z.Z. *et al*. (2004) iProLINK: an integrated protein resource for literature mining. *Comput. Biol. Chem*., **28**, 409–416.

Hu,Z.Z. *et al*. (2005) Literature mining and database annotation of protein phosphorylation using a rule-based system. *Bioinformatics*, **21**, 2759–2765.

Joachims,T. (1998) Text categorization with support vector machines: learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning,* Chemnitz, Germany, pp. 137–142.

Joachims,T. (1999) Making large-scale SVM learning practical. In *Advances in Kernel Methods—Support Vector Learning*. In Schölkopf,B., Burges,C. and Smola,A. (eds), MIT Press, Cambridge, MA, pp. 41–54.

Marcotte,E.M. *et al*. (2001) Mining literature for protein-protein interactions. *Bioinformatics*, **17**, 359–363.

McCallum,A. and Nigam,K. (1998) A comparison of event models for Naïve Bayes text classification. In *Proceedings of the AAAI-98 Workshop on Learning for Text Categorization,* Madison, WI, pp. 41–48.

Nenadic,G. *et al*. (2003) Selecting text features for gene name classification: from documents to terms. In *Proceedings of ACL 2003 Workshop on Natural Language Processing in Biomedicine,* Sapporo, Japan, pp. 121–128.

Porter,M. (1980) An algorithm for sux stripping. *Program*, **14**, 130–137.

Regev,Y. *et al*. (2003) Rulebased extraction of experimental evidence in the biomedical domain—the KDD Cup 2002 (task 1). *SIGKDD Explor. Newslett*., **4**, 90–92.

Rice,S.B. *et al*. (2005) Mining protein function from text using term-based support vector machines. *BMC Bioinformatics*, **6** (Suppl. 1), S22.

Shi,M. *et al*. (2003) A machine learning approach for the curation of biomedical literature–KDD Cup 2002 (task 1). *SIGKDD Explor. Newslett*., **4**, 93–94.

Vapnik,V.N. (1995) *The Nature of Statistical Learning Theory*. Springer, New York.

Wilbur,J.W. (2000) Boosting Naïve Bayesian learning on a large subset of MEDLINE. In *Proceedings of AMIA Symposium,* Los Angeles, CA, pp. 918–922.

Wu,C.H. *et al*. (2003) The Protein information resource. *Nucleic Acids Res*., **31**, 345–347.

Wu,C.H. *et al*. (2006) The Universal Protein Resource (UniProt): an expanding universe of protein information. *Nucleic Acids Res*., **34**, D187–D191.

Yang,Y. and Pederson,J.O. (1997) A comparative study on feature selection in text categorization. In *Proceedings of the 14th International Conference on Machine Learning*, Nashville, TN, pp. 412–420.