This is a pre-print of a paper that will appear in the Proceedings of the BioNLP Workshop at the Human Language Technology conference - North American chapter of the Association for Computational Linguistics (HLT-NAACL), held June 2006 and it will be published by ACM.

Term Generalization and Synonym Resolution for Biological Abstracts: Using the Gene Ontology for Subcellular Localization Prediction

Alona Fyshe
Department of Computing Science
University of Alberta
Edmonton, Alberta T6G 2E8
alona@cs.ualberta.ca

Duane Szafron
Department of Computing Science
University of Alberta
Edmonton, Alberta T6G 2E8
duane@cs.ualberta.ca

Abstract

The field of molecular biology is growing at an astounding rate and research findings are being deposited into public databases, such as Swiss-Prot. Many of the over 200,000 protein entries in Swiss-Prot 49.1 lack annotations such as subcellular localization or function, but the vast majority have references to journal abstracts describing related research. These abstracts represent a huge amount of information that could be used to generate annotations for proteins automatically. Training classifiers to perform text categorization on abstracts is one way to accomplish this task. We present a method for improving text classification for biological journal abstracts by generating additional text features using the knowledge represented in a biological concept hierarchy (the Gene Ontology). The structure of the ontology, as well as the synonyms recorded in it, are leveraged by our simple technique to significantly improve the F-measure of subcellular localization text classifiers by as much as 0.077 and we achieve F-measures as high as 0.935.

1 Introduction

Can computers extract the semantic content of academic journal abstracts? This paper explores the use of natural language techniques for processing biological abstracts to answer this question in a specific domain. Our prototype method predicts the subcellular localization of proteins (the part of the biological cell where a protein performs its function) by performing text classification on related journal abstracts.

In the last two decades, there has been explosive growth in molecular biology research. Molecular biologists organize their findings into a common set of databases. One such database is Swiss-Prot, in which each entry corresponds to a protein. As of version 49.1 (February 21, 2006) Swiss-Prot contains more than 200,000 proteins, 190,000 of which link to biological journal abstracts. Unfortunately, a much smaller percentage of protein entries are annotated with other types of information. For example, only about half the entries have subcellular localization annotations. This disparity is partially due to the fact that humans annotate these databases manually and cannot keep up with the influx of data. If a computer could be trained to produce annotations by processing journal abstracts, proteins in the Swiss-Prot database could be curated semi-automatically.

Document classification is the process of categorizing a set of text documents into one or more of a predefined set of classes. The classification of biological abstracts is an interesting specialization of general document classification, in that scientific language is often not understandable by, nor written for, the lay-person. It is full of specialized terms, acronyms and it often displays high levels of synonymy. For example, the "PAM complex", which exists in the mitochondrion of the biological cell is also referred to with the phrases "presequence translocase-associated import motor" and

"mitochondrial import motor". This also illustrates the fact that biological terms often span word boundaries and so their collective meaning is lost when text is whitespace tokenized.

To overcome the challenges of scientific language, our technique employs the Gene Ontology (GO) (Ashburner et al., 2000) as a source of expert knowledge. The GO is a controlled vocabulary of biological terms developed and maintained by biologists. In this paper we use the knowledge represented by the GO to complement the information present in journal abstracts. Specifically we show that:

- the GO can be used as a thesaurus
- the hierarchical structure of the GO can be used to generalize specific terms into broad concepts
- simple techniques using the GO significantly improve text classification

Although biological abstracts are challenging documents to classify, solving this problem will yield important benefits. With sufficiently accurate text classifiers, the abstracts of Swiss-Prot entries could be used to automatically annotate corresponding proteins, meaning biologists could more efficiently identify proteins of interest. Less time spent sifting through unannotated proteins translates into more time spent on new science, performing important experiments and uncovering fresh knowledge.

2 Related Work

Several different learning algorithms have been explored for text classification (Dumais et al., 1998) and support vector machines (SVMs) (Vapnik, 1995) were found to be the most computationally efficient and to have the highest precision/recall break-even point (BEP, the point where precision equals recall). Joachims performed a very thorough evaluation of the suitability of SVMs for text classification (Joachims, 1998). Joachims states that SVMs are perfect for textual data as it produces sparse training instances in very high dimensional space.

Soon after Joachims' survey, researchers started using SVMs to classify biological journal abstracts. Stapley et al. (2002) used SVMs to predict the subcellular localization of yeast proteins. They created a data set by mining Medline for abstracts containing a yeast gene name, which achieved F-measures in the range [0.31-0.80]. F-measure is defined as

$$f = \frac{2rp}{r+p}$$

where p is precision and r is recall. They expanded their training data to include extra biological information about each protein, in the form of amino acid content, and raised their F-measure by as much as 0.05. These results are modest, but before Stapley et al. most localization classification systems were built using text rules or were sequence based. This was one of the first applications of SVMs to biological journal abstracts and it showed that text and amino acid composition together yield better results than either alone.

Properties of proteins themselves were again used to improve text categorization for animal, plant and fungi subcellular localization data sets (Höglund et al., 2006). The authors' text classifiers were based on the most distinguishing terms of documents, and they included the output of four protein sequence classifiers in their training data. They measure the performance of their classifier using what they call sensitivity and specificity, though the formulas cited are the standard definitions of recall and precision. Their text-only classifier for the animal MultiLoc data set had recall (sensitivity) in the range [0.51-0.93] and specificity (precision) [0.32-0.91]. The MultiLocText classifiers, which include sequence-based classifications, have recall [0.82-0.93] and precision [0.55-0.95]. Their overall and average accuracy increased by 16.2% and 9.0% to 86.4% and 94.5% respectively on the PLOC animal data set when text was augmented with additional sequence-based information.

Our method is motivated by the improvements that Stapley et al. and Höglund et al. saw when they included additional biological information. However, our technique uses knowledge of a textual nature to improve text classification; it uses no information from the amino acid sequence. Thus, our approach can be used in conjunction with techniques that use properties of the protein sequence.

In non-biological domains, external knowledge has already been used to improve text categorization (Gabrilovich and Markovitch, 2005). In their

research, text categorization is applied to news documents, newsgroup archives and movie reviews. The authors use the Open Directory Project (ODP) as a source of world knowledge to help alleviate problems of polysemy and synonymy. The ODP is a hierarchy of concepts where each concept node has links to related web pages. The authors mined these web pages to collect characteristic words for each concept. Then a new document was mapped, based on document similarity, to the closest matching ODP concept and features were generated from that concept's meaningful words. The generated features, along with the original document, were fed into an SVM text classifier. This technique yielded BEP as high as 0.695 and improvements of up to 0.254.

We use Gabrilovich and Markovitch's (2005) idea to employ an external knowledge hierarchy, in our case the GO, as a source of information. It has been shown that GO molecular function annotations in Swiss-Prot are indicative of subcellular localization annotations (Lu and Hunter, 2005), and that GO node names made up about 6% of a sample Medline corpus (Verspoor et al., 2003). Some consider GO terms to be too rare to be of use (Rice et al., 2005), however we will show that although the presence of GO terms is slight, the terms are powerful enough to improve text classification. Our technique's success may be due to the fact that we include the synonyms of GO node names, which increases the number of GO terms found in the documents.

We use the GO hierarchy in a different way than Gabrilovich et al. use the ODP. Unlike their approach, we do not extract additional features from all articles associated with a node of the GO hierarchy. Instead we use synonyms of nodes and the names of ancestor nodes. This is a simpler approach, as it doesn't require retrieving all abstracts for all proteins of a GO node. Nonetheless, we will show that our approach is still effective.

3 Methods

The workflow used to perform our experiments is outlined in Figure 1.

3.1 The Data Set

The first step in evaluating the usefulness of GO as a knowledge source is to create a data set. This process begins with a set of proteins with known subcellular localization annotations (Figure 1). For this we use Proteome Analyst's (PA) data sets (Lu et al., 2004; Szafron et al., 2004). The PA group used these data sets to create very accurate subcellular classifiers based on the keyword fields of Swiss-Prot entries for homologous proteins. Here we use PA's current data set of proteins collected from Swiss-Prot (version 48.3) and impose one further criterion: the subcellular localization annotation may not be longer than four words. This constraint is introduced to avoid including proteins where the localization category was incorrectly extracted from a long sentence describing several aspects of localization. For example, consider the subcellular annotation "attached to the plasma membrane by a lipid anchor", which could mean the protein's functional components are either cytoplasmic or extracellular (depending on which side of the plasma membrane the protein is anchored). PA's simple parsing scheme could mistake this description as meaning that the protein performs its function in the plasma membrane. Our length constraint reduces the chances of including mislabeled training instances in our data.

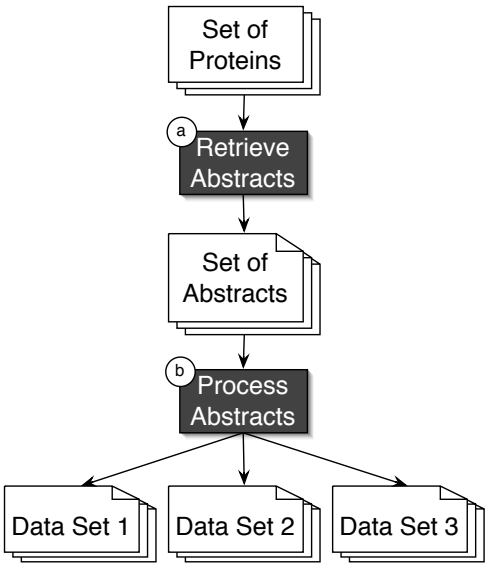

Figure 1: The workflow used to create data sets used in this paper. Abstracts are gathered for proteins with known localization (process *a*). Treatments are applied to abstracts to create three Data Sets (process *b*).

Class Name	Number of Proteins	Number of Abstracts
cytoplasm	1664	4078
endoplasmic reticulum	310	666
extracellular	2704	5655
golgi [a]	41	71
lysosome	129	599
mitochondrion	559	1228
nucleus	2445	5589
peroxisome	108	221
plasma membrane [a]	15	38
Total	**7652**	**17175**

[a]Classes with less than 100 abstracts were considered to have too little training data and are not included in our experiments.

Table 1: Summary of our Data Set. Totals are less than the sum of the rows because proteins may belong to more than one localization class.

PA has data sets for five organisms (animal, plant, fungi, gram negative bacteria and gram positive bacteria). The animal data set was chosen for our study because it is PA's largest and medical research has the most to gain from increased annotations for animal proteins. PA's data sets have binary labeling, and each class has its own training file. For example, in the nuclear data set a nuclear protein appears with the label "$+1$", and non-nuclear proteins appear with the label "-1". Our training data includes 317 proteins that localize to more than one location, so they will appear with a positive label in more than one data set. For example, a protein that is both cytoplasmic and peroxisomal will appear with the label "$+1$" in both the peroxisomal and cytoplasmic sets, and with the label "-1" in all other sets. Our data set has 7652 proteins across 9 classes (Table 1). To take advantage of the information in the abstracts of proteins with multiple localizations, we use a one-against-all classification model, rather than a "single most confident class" approach.

3.2 Retrieve Abstracts

Now that a set of proteins with known localizations has been created, we gather each protein's abstracts and abstract titles (Figure 1, process a). We do not include full text because it can be difficult to obtain automatically and because using full text does not improve F-measure (Sinclair and Webber, 2004). Abstracts for each protein are retrieved using the PubMed IDs recorded in the Swiss-Prot database. PubMed (`http://www.pubmed.gov`) is a database of life science articles. It should be noted that more than one protein in Swiss-Prot may point to the same abstract in PubMed. Because the performance of our classifiers is estimated using cross-validation (discussed in Section 3.4) it is important that the same abstract does not appear in both testing and training sets during any stage of cross-validation. To address this problem, all abstracts that appear more than once in the complete set of abstracts are removed. The distribution of the remaining abstracts among the 9 subcellular localization classes is shown in Table 1. For simplicity, the fact that an abstract may actually be discussing more than one protein is ignored. However, because we remove duplicate abstracts, many abstracts discussing more than one protein are eliminated.

In Table 1 there are more abstracts than proteins because each protein may have more than one associated abstract. Classes with less than 100 abstracts were deemed to have too little information for training. This constraint eliminated plasma membrane and golgi classes, although they remained as negative data for the other 7 training sets.

It is likely that not every abstract associated with a protein will discuss subcellular localization. However, because the Swiss-Prot entries for proteins in our data set have subcellular annotations, some research must have been performed to ascertain localization. Thus it should be reported in at least one abstract. If the topics of the other abstracts are truly unrelated to localization than their distribution of words may be the same for all localization classes. However, even if an abstract does not discuss localization directly, it may discuss some other property that is correlated with localization (e.g. function). In this case, terms that differentiate between localization classes will be found by the classifier.

3.3 Processing Abstracts

Three different data sets are made by processing our retrieved abstracts (Figure 1, process b). An ex-

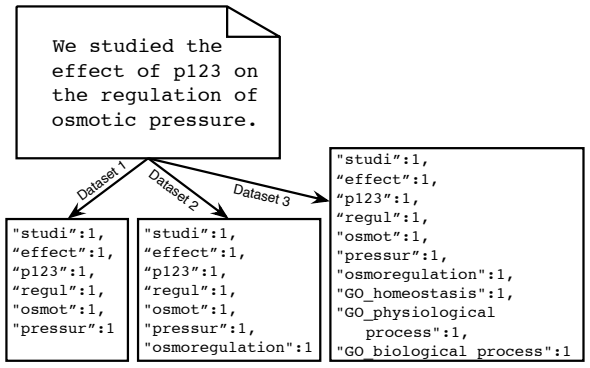

Figure 2: A sentence illustrating our three methods of abstract processing. Data Set 1 is our baseline, Data Set 2 incorporates synonym resolution and Data Set 3 incorporates synonym resolution and term generalization. Word counts are shown here for simplicity, though our experiments use TFIDF.

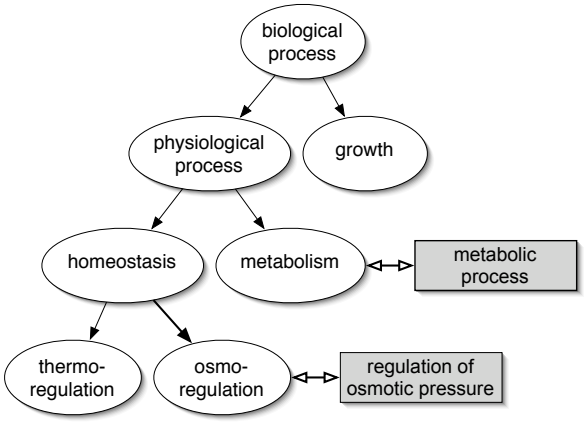

Figure 3: A subgraph of the GO biological process hierarchy. GO nodes are shown as ovals, synonyms appear as grey rectangles.

ample illustrating our three processing techniques is shown in Figure 2.

In Data Set 1, abstracts are tokenized and each word is stemmed using Porter's stemming algorithm (Porter, 1980). The words are then transformed into a vector of <word,TFIDF> pairs. TFIDF is defined as:

$$TFIDF(w_i) = f(w_i) * log(\frac{n}{D(w_i)})$$

where $f(w_i)$ is the number of times word w_i appears in documents associated with a protein, n is the total number of training documents and $D(w_i)$ is the number of documents in the whole training set that contain the word w_i. TFIDF was first proposed by Salton and Buckley (1998) and has been used extensively in various forms for text categorization (Joachims, 1998; Stapley et al., 2002). The words from all abstracts for a single protein are amalgamated into one "bag of words" that becomes the training instance which represents the protein.

3.3.1 Synonym Resolution

The GO hierarchy can act as a thesaurus for words with synonyms. For example the GO encodes the fact that "metabolic process" is a synonym for "metabolism"(see Figure 3). Data Set 2 uses GO's "exact_synonym" field for synonym resolution and adds extra features to the vector of words from Data Set 1. We search a stemmed version of the abstracts for matches to stemmed GO node names or synonyms. If a match is found, the GO node name (deemed the canonical representative for its set of synonyms) is associated with the abstract. In Figure 2 the phrase "regulation of osmotic pressure" appears in the text. A lookup in the GO synonym dictionary will indicate that this is an exact synonym of the GO node "osmoregulation". Therefore we associated the term "osmoregulation" with the training instance. This approach combines the weight of several synonyms into one representative, allowing the SVM to more accurately model the author's intent, and identifies multi-word phrases that are otherwise lost during tokenization. Table 2 shows the increase in average number of features per training instance as a result of our synonym resolution technique.

3.3.2 Term Generalization

In order to express the relationships between terms, the GO hierarchy is organized in a directed acyclic graph (DAG). For example, "thermoregulation" is a type of "homeostasis", which is a "physiological process". This "is a" relationship is expressed as a series of parent-child relationships (see Figure 3). In Data Set 3 we use the GO for synonym resolution (as in Data Set 2) and we also use its hierarchical structure to generalize specific terms into broader concepts. For Data Set 3, if a GO node name (or synonym) is found in an abstract, all names of ancestors to the match in the text are included in the

Class	Data Set 1	Data Set 2	Data Set 3
cytoplasm	166	177	203
endoplasmic reticulum	162	171	192
extracellular	148	155	171
lysosome	244	255	285
mitochondrion	155	163	186
nucleus	147	158	183
peroxisome	147	156	182
Overall Average	167	176	200

Table 2: Average number of features per training instance for 7 subcellular localization categories in animals. Data Set 1 is the baseline, Data Set 2 incorporates synonym resolution and Data Set 3 uses synonym resolution and term generalization.

training instance along with word vectors from Data Set 2 (see Figure 2). These additional node names are prepended with the string "GO_" which allows the SVM to differentiate between the case where a GO node name appears exactly in text and the case where a GO node name's child appeared in the text and the ancestor was added by generalization. Term generalization increases the average number of features per training instance (Table 2).

Term generalization gives the SVM algorithm the opportunity to learn correlations that exist between general terms and subcellular localization even if the general term never appears in an abstract and we encounter only its more specific children. Without term generalization the SVM has no concept of the relationship between child and parent terms, nor between sibling terms. For some localization categories more general terms may be the most informative and in other cases specific terms may be best. Because our technique adds features to training instances and never removes any, the SVM can assign lower weights to the generalized terms in cases where the localization category demands it.

3.4 Evaluation

Each of our classifiers was evaluated using 10 fold cross-validation. In 10 fold cross-validation each Data Set is split into 10 stratified partitions. For the first "fold", a classifier is trained on 9 of the 10 partitions and the tenth partition is used to test the classifier. This is repeated for nine more folds, holding out a different tenth each time. The results of all 10 folds are combined and composite precision, recall and F-measures are computed. Cross-validation accurately estimates prediction statistics of a classifier, since each instance is used as a test case at some point during validation.

The SVM implementation libSVM (Chang and Lin, 2001) was used to conduct our experiments. A linear kernel and default parameters were used in all cases; no parameter searching was done. Precision, recall and F-measure were calculated for each experiment.

4 Results and Discussion

Results of 10 fold cross-validation are reported in Table 3. Data Set 1 represents the baseline, while Data Sets 2 and 3 represent synonym resolution and combined synonym resolution/term generalization respectively. Paired t-tests (p=0.05) were done between the baseline, synonym resolution and term generalization Data Sets, where each sample is one fold of cross-validation. Those classifiers with significantly better performance over the baseline appear in bold in Table 3. For example, the lysosome classifiers trained on Data Set 2 and 3 are both significantly better than the baseline, and results for Data Set 3 are significantly better than results for Data Set 2, signified with an asterisk. In the case of the nucleus classifier no abstract processing technique was significantly better, so no column appears in bold.

In six of the seven classes, classifiers trained on Data Set 2 are significantly better than the baseline, and in no case are they significantly worse. In Data Set 3 five of the seven classifiers are significantly better than the baseline, and in no case are they significantly worse. Although our results for nucleus' Data Set 3 dropped by 0.7% this decrease in accuracy is not significant. For the lysosome and peroxisome classes our combined synonym resolution/term generalization technique produced results that are significantly better than synonym resolution alone. The average results of Data Set 2 are significantly better than Data Set 1 and the average results of Data Set 3 are significantly better than Data

Class	**Data Set 1** Baseline	**Data Set 2** Synonym Resolution		**Data Set 3** Term Generalization	
	F-measure	F-Measure	Δ	F-Measure	Δ
cytoplasm	0.739 (±0.049)	**0.756** (±0.042)	+0.017	**0.760** (±0.042)	+0.021
endoplasmic reticulum	0.759 (±0.055)	**0.779** (±0.068)	+0.020	**0.779** (±0.072)	+0.020
extracellular	0.931 (±0.009)	**0.935** (±0.009)	+0.004	**0.935** (±0.010)	+0.004
lysosome	0.740 (±0.107)	**0.782** (± 0.100)	+0.042	**0.817*** (±0.089)	+0.077
mitochondrion	0.839 (±0.041)	**0.848** (±0.038)	+0.009	0.851 (±0.039)	+0.012
nucleus	0.884 (±0.014)	0.885 (± 0.016)	+0.001	0.877 (±0.019)	-0.007
peroxisome	0.790 (±0.054)	**0.822** (±0.042)	+0.032	**0.867*** (±0.046)	+0.077
Average	0.812 (±0.016)	**0.830** (±0.012)	+0.018	**0.843*** (±0.009)	+0.031

Table 3: F-measures for stratified 10 fold cross-validation on our three Data Sets. Results deemed significantly improved over the baseline (p=0.05) appear in **bold**, and those with an asterisk (*) are significantly better than both other data sets. Change in F-measure compared to baseline is shown for Data Sets 2 and 3. Standard deviation is shown in parentheses.

Set 2 and Data Set 1. On average, synonym resolution and term generalization combined give an improvement of 3.1%, and synonym resolution alone yields a 1.8% improvement. Because term generalization and synonym resolution never produce classifiers that are significantly worse than synonym resolution alone, and in some cases the result is 7.7% better than the baseline, Data Set 3 can be confidently used for text categorization of all seven animal subcellular localization classes.

Our baseline SVM classifier performs quite well compared to the baselines reported in related work. At worst, our baseline classifier has F-measure 0.739. The text only classifier reported by Höglund et al. has F-measure in the range [0.449,0.851] (Höglund et al., 2006) and the text only classifiers presented by Stapley et al. begin with a baseline classifier with F-measure in the range [0.31,0.80] (Stapley et al., 2002). Although their approaches gave a greater increase in performance their low baselines left more room for improvement.

The reader should keep in mind that our data sets differ from those used by Höglund et al. (2006) when considering the following comparison to their work. For those 7 localization classes for which we both make predictions, the F-measure of our classifiers trained on Data Set 3 exceed the F-measures of the Höglund et al. text only classifiers in all cases, and our Data Set 3 classifier beats the F-measure of the MutliLocText classifier for 5 classes (see supplementary material `http://www.cs.ualberta.ca/~alona/bioNLP`). In addition, our technique does not preclude using techniques presented by Höglund et al. and Stapley et al., and it may be that using a combination of our approach and techniques involving protein sequence information may result in an even stronger subcellular localization predictor.

We do not assert that using abstract text alone is the best way to predict subcellular localization, only that if text is used, one must extract as much from it as possible. We are currently working on incorporating the classifications given by our text classifiers into Proteome Analyst's subcellular classifier to improve upon its already strong predictors (Lu et al., 2004), as they do not currently use any information present in the abstracts of homologous proteins.

5 Conclusion and Future work

Our study has shown that using an external information source is beneficial when processing abstracts from biological journals. The GO can be used as a reference for both synonym resolution and term generalization for document classification and doing so significantly increases the F-measure of most subcellular localization classifiers for animal proteins. On average, our improvements are modest, but they indicate that further exploration of this technique is

warranted.

We are currently repeating our experiments for PA's other subcellular data sets and for function prediction. Though our previous work with PA is not text based, our experience training protein classifiers has led us to believe that a technique that works well for one protein property often succeeds for others as well. For example our general function classifier has F-measure within one percent of the F-measure of our Animal subcellular classifier. Although we test the technique presented here on subcellular localization only, we see no reason why it could not be used to predict any protein property (general function, tissue specificity, relation to disease, etc.). Finally, although our results apply to text classification for molecular biology, the principle of using an ontology that encodes synonyms and hierarchical relationships may be applicable to other applications with domain specific terminology.

The Data Sets used in these experiments are available at `http://www.cs.ualberta.ca/~alona/bioNLP/`.

6 Acknowledgments

We would like to thank Greg Kondrak, Colin Cherry, Shane Bergsma and the whole NLP group at the University of Alberta for their helpful feedback and guidance. We also wish to thank Paul Lu, Russell Greiner, Kurt McMillan and the rest of the Proteome Analyst team. This research was made possible by financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC), the Informatics Circle of Research Excellence (iCORE) and the Alberta Ingenuity Centre for Machine Learning (AICML).

References

Michael Ashburner et al. 2000. Gene ontology: tool for the unification of biology the gene ontology consortium. *Nature Genetics*, 25(1):25–29.

Chih-Chung Chang and Chih-Jen Lin, 2001. *LIBSVM: a library for support vector machines*. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Susan T. Dumais et al. 1998. Inductive learning algorithms and representations for text categorization. In *Proc. 7th International Conference on Information and Knowledge Management CIKM*, pages 148–155.

Evgeniy Gabrilovich and Shaul Markovitch. 2005. Feature generation for text categorization using world knowledge. In *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence*, pages 1048–1053.

Annette Höglund et al. 2006. Significantly improved prediction of subcellular localization by integrating text and protein sequence data. In *Pacific Symposium on Biocomputing*, pages 16–27.

Thorsten Joachims. 1998. Text categorization with suport vector machines: Learning with many relevant features. In *ECML '98: Proceedings of the 10th European Conference on Machine Learning*, pages 137–142.

Zhiyong Lu and Lawrence Hunter. 2005. GO molecular function terms are predictive of subcellular localization. volume 10, pages 151–161.

Zhiyong Lu et al. 2004. Predicting subcellular localization of proteins using machine-learned classifiers. *Bioinformatics*, 20(4):547–556.

Martin F. Porter. 1980. An algorithm for suffix stripping. *Program*, 14(3):130–137.

Simon B Rice et al. 2005. Mining protein function from text using term-based support vector machines. *BMC Bioinformatics*, 6:S22.

Gail Sinclair and Bonnie Webber. 2004. Classification from full text: A comparison of canonical sections of scientific papers. In *COLING 2004 International Joint workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP) 2004*, pages 69–72.

B. J. Stapley et al. 2002. Predicting the sub-cellular location of proteins from text using support vector machines. In *Pacific Symposium on Biocomputing*, pages 374–385.

Duane Szafron et al. 2004. Proteome analyst: Custom predictions with explanations in a web-based tool for high-throughput proteome annotations. *Nucleic Acids Research*, 32:W365–W371.

Vladimir N Vapnik. 1995. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., New York, NY, USA.

Cornelia M. Verspoor et al. 2003. The gene ontology as a source of lexical semantic knowledge for a biological natural language processing application. *Proceedings of the SIGIR'03 Workshop on Text Analysis and Search for Bioinformatics*.