

Support Vector Machine Approach to Extracting Gene References into Function from Biological Documents

Chih Lee, Wen-Juan Hou and Hsin-Hsi Chen

Natural Language Processing Laboratory
Department of Computer Science and Information Engineering
National Taiwan University
1 Roosevelt Road, Section 4, Taipei, Taiwan, 106
{clee, wjhou}@nlg.csie.ntu.edu.tw, hh_chen@csie.ntu.edu.tw

Abstract

In the biological domain, extracting newly discovered functional features from the massive literature is a major challenging issue. To automatically annotate Gene References into Function (GeneRIF) in a new literature is the main goal of this paper. We tried to find GRIF words in a training corpus, and then applied these informative words to annotate the GeneRIFs in abstracts with several different weighting schemes. The experiments showed that the Classic Dice score is at most 50.18%, when the weighting schemes proposed in the paper (Hou *et al.*, 2003) were adopted. In contrast, after employing Support Vector Machines (SVMs) and the definition of classes proposed by Jelier *et al.* (2003), the score greatly improved to 56.86% for Classic Dice (CD). Adopting the same features, SVMs demonstrated advantage over the Naïve Bayes Classifier. Finally, the combination of the former two models attained a score of 59.51% for CD.

1 Introduction

Text Retrieval Conference (TREC) has been dedicated to information retrieval and information extraction for years. TREC 2003 introduced a new track called Genomics Track (Hersh and Bhupatiraju, 2003) to address the information retrieval and information extraction issues in the biomedical domain. For the information extraction part, the goal was to automatically reproduce the Gene Reference into Function (GeneRIF) resource in the LocusLink database (Pruitt *et al.*, 2000.) GeneRIF associated with a gene is a sentence describing the function of that gene, and is currently manually generated.

This paper presents the post-conference work on the information extraction task (i.e., secondary task). In the official runs, our system (Hou *et al.*, 2003) adopted several weighting schemes (described in Section 3.2) to deal with this problem. However,

we failed to beat the simple baseline approach, which always picks the title of a publication as the candidate GeneRIF. Bhalotia *et al.* (2003) converted this task into a binary classification problem and trained a Naïve Bayes classifier with kernels, achieving 53.04% for CD. In their work, the title and last sentence of an abstract were concatenated and features were then extracted from the resulting string. Jelier *et al.* (2003) observed the distribution of target GeneRIFs in 9 sentence positions and converted this task into a 9-class classification problem, attaining 57.83% for CD. Both works indicated that the sentence position is of great importance. We therefore modified our system to incorporate the position information with the help of SVMs and we also investigated the capability of SVMs versus Naïve Bayes on this problem.

The rest of this paper is organized as follows. Section 2 presents the architecture of our extracting procedure. The basic idea and the experimental methods in this study are introduced in Section 3. Section 4 shows the results and makes some discussions. Finally, Section 5 concludes the remarks and lists some future works.

2 Architecture Overview

A complete annotation system may be done at two stages, including (1) extraction of molecular function for a gene from a publication and (2) alignment of this function with a GO term. Figure 1 shows an example. The left part is an MEDLINE abstract with the function description highlighted. The middle part is the corresponding GeneRIF. The matching words are in bold, and the similar words are underlined. The right part is the GO annotation. This figure shows a possible solution of maintaining the knowledge bases and ontology using natural language processing technology. We addressed automation of the first stage in this paper.

The overall architecture is shown in Figure 2. First, we constructed a training corpus in such a way that GeneRIFs were collected from LocusLink and the corresponding abstracts were retrieved from

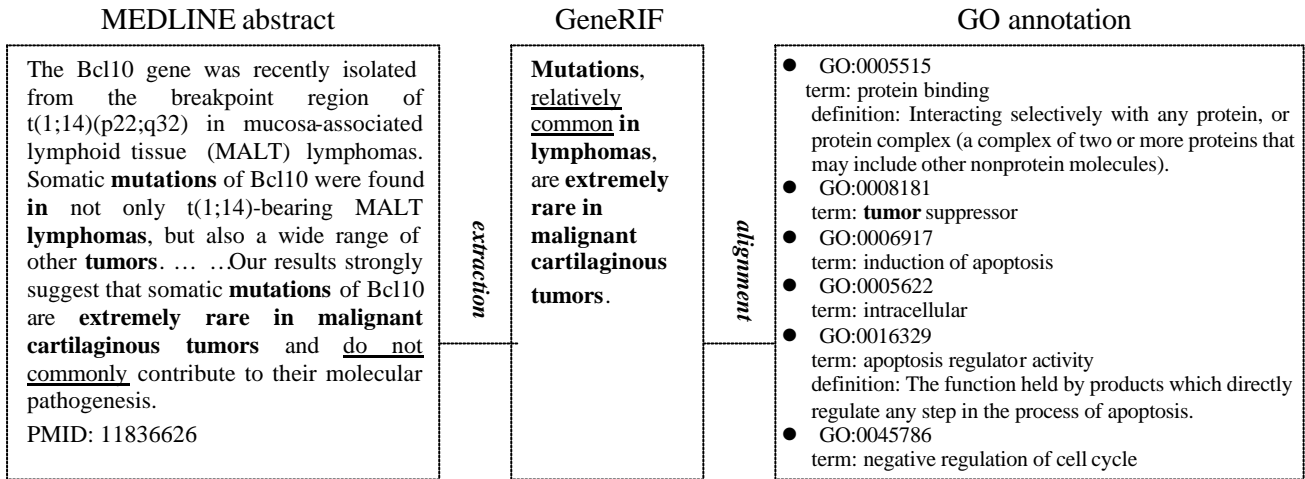


Figure 1: An Example of Complete Annotation from a Literature to Gene Ontology

MEDLINE. “GRIF words” and their weights were derived from the training corpus. Then Support Vector Machines were trained using the derived corpus. Given a new abstract, a sentence is selected from the abstract to be the candidate GeneRIF.

3 Methods

We adopted several weighting schemes to locate the GeneRIF sentence in an abstract in the official runs (Hou *et al.*, 2003). Inspired by the work by Jelier *et al.* (2003), we incorporated their definition of classes into our weighting schemes, converting this task into a classification problem using SVMs as the classifier. We ran SVMs on both sets of features proposed by Hou *et al.* (2003) and Jelier *et al.* (2003), respectively. Finally, all the features were combined and some feature selection methods were applied to train the classifier.

3.1 Training and test material preparation

Since GeneRIFs are often cited verbatim from abstracts, we decided to reproduce the GeneRIF by selecting one sentence in the abstract. Therefore, for each abstract in our training corpus, the sentence most similar to the GeneRIF was labelled as the GeneRIF sentence using Classic Dice coefficient as similarity measure. Totally, 259,244 abstracts were

used, excluding the abstracts for testing. The test data for evaluation are the 139 abstracts used in TREC 2003 Genomics track.

3.2 GRIF words extraction and weighting scheme

We called the matched words between GeneRIF and the selected sentence as *GRIF words* in this paper. GRIF words represent the favorite vocabulary that human experts use to describe gene functions. After stop word removal and stemming operation, 10,506 GRIF words were extracted.

In our previous work (Hou *et al.*, 2003), we first generated the weight for each GRIF word. Given an abstract, the score of each sentence is the sum of weights of all the GRIF words in this sentence. Finally, the sentence with the highest score is selected as the candidate GeneRIF. This method is denoted as OUR weighting scheme, and several heuristic weighting schemes were investigated. Here, we only present the weighting scheme used in SVMs classification. The weighting scheme is as follows. For GRIF word i , the number of occurrence n_i^G in all the GeneRIF sentences and the number of occurrence n_i^A in all the abstracts were computed and n_i^G / n_i^A was assigned to GRIF word i as its weight.

3.3 Classification

3.3.1 Class definition and feature extraction

The distribution of GeneRIF sentences showed that the position of a sentence in an abstract is an important clue to where the answer sentence is. Jelier *et al.* (2003) considered only the title, the first three and the last five sentences, achieving the best performance in TREC official runs. Their Naïve Bayes model is as follows. An abstract a is assigned a class v_j by calculating v_{NB} :

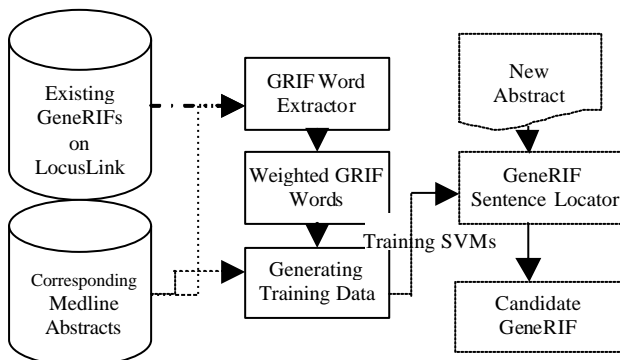


Figure 2: Architecture of Extracting Candidate GeneRIF

$$v_{NB} = \underset{v_j \in V}{\operatorname{argmax}} P(v_j) \times \prod_{i \in S} \prod_{k \in W_{a,i}} P(w_{k,i} | v_j)$$

where v_j is one of the nine positions aforementioned, S is the set of 9 sentence positions, $W_{a,i}$ is the set of all word positions in sentence i in abstract a , $w_{k,i}$ is the occurrence of the normalized word at position k in sentence i and V is the set of 9 classes.

We, therefore, represented each abstract by a feature vector composed of the scores of 9 sentences. Furthermore, with a list of our 10,506 GRIF words at hand, we also computed the occurrences of these words in each sentence, given an abstract. Each abstract is then represented by the number of occurrences of these words in the 9 sentences respectively, i.e., the feature vector is 94,554 in length. Classification based on this type of features is denoted the *sentence-wise bag of words model* in the rest of this paper. Combining these two models, we got totally 94,563 features.

Since we are extracting sentences discussing gene functions, it's reasonable to expect gene or protein names in the GeneRIF sentence. Therefore, we employed Yapex (Olsson *et al.*, 2002) and GAPSCORE (Chang *et al.*, 2004) protein/gene name detectors to count the number of protein/gene names in each of the 9 sentences, resulting in 94,581 features.

3.3.2 Training SVMs

The whole process related to SVM was done via LIBSVM – A Library for Support Vector Machines (Hsu *et al.*, 2003). Radial basis kernel was adopted based on our previous experience. However, further verification showed that the combined model with either linear or polynomial kernel only slightly surpassed the baseline, attaining 50.67% for CD. In order to get the best-performing classifier, we tuned two parameters, C and gamma. They are the penalty coefficient in optimization and a parameter for the radial basis kernel, respectively. Four-fold cross validation accuracy was used to select the best parameter pair.

3.3.3 Picking up the answer sentence

Test instances were first fed to the classifier to get the predicted positions of GeneRIF sentences. In case that the predicted position doesn't have a sentence, which would happen when the abstract doesn't have enough sentences, the sentence with the highest score is picked for the weighting scheme and the combined model, otherwise the title is picked for the sentence-wise bag of words model.

4 Results and Discussions

The performance measures are based on Dice coefficient, which calculates the overlap between the candidate GeneRIF and actual GeneRIF. Classic Dice (CD) is the classic Dice formula using a common stop word list and the Porter stemming algorithm. Due to lack of space, we referred you to the Genomics track overview for the other three modifications of CD (Hersh and Bhupatiraju, 2003).

The evaluation results are shown in Table 2. The 1st row shows the official run of Jelier's team, the first place in the official runs. The 2nd row shows the performance when the Naïve Bayes classifier adopted by Jelier is replaced with SVMs. The 3rd row is the performance of our weighting scheme without a classifier. The 4th row then lists the performance when our weighting scheme is combined with SVMs. The 5th row is the result when our weighting scheme and the sentence-wise bag of words model are combined together. The 6th row is the result when two gene/protein name detectors are incorporated into the combined model. The next two rows were obtained after two feature selection methods were applied. The 9th row shows the performance when the classifier always proposes a sentence most similar to the actual GeneRIF. The last row lists the baseline, i.e., title is always picked.

A comparative study on text categorization (Joachims, 1998) showed that SVMs outperform other classification methods, such as Naïve Bayes, C4.5, and k-NN. The reasons would be that SVMs are capable of handling large feature space, text categorization has few irrelevant features, and document vectors are sparse. The comparison

		CD	MUD	MBD	MBDP
1	Jelier (Sentence-wise bag of words + Naïve Bayes)	57.83%	59.63%	46.75%	49.11%
2	Sentence-wise bag of words + SVMs	58.92%	61.46%	47.86%	50.84%
3	OUR Weighting scheme	50.18%	46.71%	33.47%	38.83%
4	OUR Weighting scheme + SVMs	56.86%	58.81%	45.08%	48.10%
5	Combined	59.51%	62.16%	48.17%	51.25%
6	Combined + gene/protein names	57.59%	59.95%	46.69%	49.68%
7	Combined + BWRatio feature selection	57.59%	59.90%	47.11%	50.08%
8	Combined + Graphical feature selection	58.81%	61.09%	47.98%	50.92%
9	Optimal Classifier	67.60%	70.74%	59.28%	62.09%
10	Baseline	50.47%	52.60%	34.82%	37.91%

Table 2: Comparison of performances on the 139 abstracts

between SVMs and the Naïve Bayes classifier again demonstrated the superiority of SVMs in text categorization (rows 1, 2).

The performance greatly improved after introducing position information (rows 3, 4), showing the sentence position plays an important role in locating the GeneRIF sentence. The 2% difference between rows 2 and 4 indicates that the features under sentence-wise bag of words model are more informative than those under our weighting scheme. However, with only 9 features, our weighting scheme with SVMs performed fairly well. Comparing the performance before and after combining our weighting scheme and the sentence-wise bag of words model (rows 2, 5 and rows 4, 5), we can infer from the performance differences that both models provide mutually exclusive information in the combined model. The result shown in row 6 indicates that the information of gene/protein name occurrences did not help identify the GeneRIF sentences in these 139 test abstracts.

We performed feature selection on the combined model to reduce the dimension of feature space. There were two methods applied: a supervised heuristic method (denoted as BWRatio feature selection in Table 2) (S. Dutoit *et al.*, 2002) and another unsupervised method (denoted as Graphical feature selection in Table 2) (Chang *et al.*, 2002). The number of features was then reduced to about 4,000 for both methods. Unfortunately, the performance did not improve after either method was applied. This may be attributed to over-fitting training data, because the cross-validation accuracies are indeed higher than those without feature selection. The result may also imply there are little irrelevant features in this case.

5 Conclusion and Future work

This paper proposed an automatic approach to locate the GeneRIF sentence in an abstract with the assistance of SVMs, reducing the human effort in updating and maintaining the GeneRIF field in the LocusLink database.

We have to admit that the 139 abstracts provided in TREC 2003 are too few to verify the performance among models, and the results based on these 139 abstracts may be slightly biased. Our next step would aim at measuring the cross-validation performances using Dice coefficient.

The syntactic information is worth exploring, since the sentences describing gene functions may share some common structural patterns. Moreover, how the weighting scheme affects the performance is also very interesting. We are currently trying to obtain a weighting scheme that can best distinguish GeneRIF sentence from non-GenerIF sentence without classifiers.

References

- G. Bhalotia, P.I. Nakov, A.S. Schwartz, and M.A. Hearst. 2003. BioText Team Report for the TREC 2003 Genomics Track. *TREC 2003 work notes*: 158-166.
- Y.C. I. Chang, H. Hsu and L.Y. Chou. 2002. Graphical Features Selection Method. *Intelligent Data Engineering and Automated Learning*, Edited by H. Yin, N. Allinson, R. Freeman, J. Keane, and S. Hubbard.
- J.T. Chang, H. Schutze, R.B. Altman. 2004. GAPSCORE: finding gene and protein names one word at a time. *Bioinformatics*, 20(2):216-225.
- S. Dutoit, Y.H. Yang, M.J. Callow and T.P. Speed. 2002. Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. *J. Amer. Statist. Assoc.* 97:77-86.
- W. Hersh and Ravi Teja Bhupatiraju. 2003. TREC Genomics Track Overview. *TREC 2003 work notes*.
- W.J. Hou, C.Y. Teng, C. Lee and H.H. Chen. 2003. SVM Approach to GeneRIF Annotation. *Proceedings of TREC 2003*.
- C.W. Hsu, C.C. Chang and C.J. Lin. 2003. A Practical Guide to Support Vector Classification. <http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html>.
- R. Jelier, M. Schuemie, C.V.E. Eijk, M. Weeber, E.V. Mulligen, B. Schijvenaars, B. Mons and J. Kors. 2003. Searching for geneRIFs: concept-based query expansion and Bayes classification. *TREC 2003 work notes*: 167-174.
- T. Joachims. 1998. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. *Proceedings of ECML-98*, 137-142.
- F. Olsson, G. Eriksson, K. Franzén, L. Asker and P. Lidén. 2002. Notions of Correctness when Evaluating Protein Name Taggers. *Proceedings of the 19th International Conference on Computational Linguistics 2002*, 765-771.
- K.D. Pruitt, K.S. Katz, H. Sicotte and D.R. Maglott. 2000. Introducing RefSeq and LocusLink: Curated Human Genome Resources at the NCBI. *Trends Genet*, 16(1):44-47.
- T. Sekimizu, H.S. Park and J. Tsujii. 1998. Identifying the Interaction Between Genes and Gene Products Based on Frequently Seen Verbs in Medline Abstracts. *Genome Information*, 9:62-71