

# A Collocation Method to Extract Biological Keywords and Its Application to Protein Name Recognition

Wen-Juan Hou

wjhou@nlg.csie.ntu.edu.tw

Hsin-Hsi Chen

hh\_chen@csie.ntu.edu.tw

*Department of Computer Science and Information Engineering  
National Taiwan University*

*No. 1, Roosevelt Road Section 4, Taipei, Taiwan 106, R.O.C.*

**Keywords:** biological keywords, collocation model, protein name recognition, *t*-test

## Abstract

*Mining biological relationships from scientific text, which facilitates automatic construction of knowledge base and finding of new information, becomes one of the emerging applications. Named entity recognition is a fundamental task in relationship mining. Traditional methods listed a pre-defined set of words to indicate protein or gene interactions by intuition. The argument is that we cannot assure if the keyword set is complete. A collocation approach not only mines biological keywords for relationship establishment, but also employs the keywords to improve performance of protein name recognition. This paper proposes a collocation approach to extract biological keywords or key phrases. Frequency, mean-and-variance, and *t*-test statistics are computed. Compound collocates are also resolved in this study. The experiments show that the performances of the *t*-test model are 76.67% and 95.45% without and with considerations of text domains, respectively. This paper suggests many useful terms that the previous literatures do not touch on. The results are valuable for the investigation of relationships between proteins, and can be integrated into genome analysis systems. In the application of protein name recognition, it enhances the precision from 70.90% to 81.94% on Yapex system.*

## 1. Introduction

With the fast development of biotechnologies, many research results have been published. The rapid growth of research in biology has generated a large scale of biological data. As a result, how to extract biological knowledge from scientific text is indispensable because the researchers have to keep up with all the new and important information. Besides, scientific documents form some sort of corpora, so statistical natural language processing technologies are useful for the exploration of bioinformatics.

There are two major issues in automatic extraction of biological knowledge [7]. One is named entity extraction that identifies gene or protein names in biological text [2, 5, 8, 13]. Some of these systems are rule-based whereas others use statistical or machine learning approaches, and both of them may utilize external domain knowledge. The other one is to find functional relationships or pathways among molecular entities. For instance, [1, 11, 16] proposed some methods to extract the information on protein-protein interactions. [14, 15] dealt with gene interactions. Furthermore, [5, 9, 12] investigated the pathway extraction.

Named entity recognition is a fundamental step to mine knowledge from biological articles.

After identifying named entities, most researches [1, 9, 10, 12, 15, 16, 17] were based on some special verbs and their related noun forms to discover molecular pathways or relationships. These pre-specified words indicate actions associated to protein or gene interactions. Blaschke, *et al.* [1] used fourteen keywords for protein-protein interactions. Ng, *et al.* [9] applied some function words for the *inhibit-activate* relationships. Sekimizu, *et al.* [15] extracted gene relations associated with seven frequently seen verbs found in MEDLINE abstracts. All these papers except Sekimizu [15] listed the keywords by intuition. Some keywords are common to most of the papers and some are special. The argument of the above approach is that we cannot assure if the keyword set is complete for mining biological relationships.

This paper aims to identify significant biological words/phrases (called keywords or key phrases later) automatically, which often accompany with molecular entities directly in the biological documents. To distinguish if a particular keyword or a key phrase co-occurs with proteins by chance, we apply the statistical *t*-test model. The mined keywords cannot only be used to construct the relationship graph between genes or proteins, but also to act as informative features in protein/gene name recognition. The rest of the paper is organized as follows. The biological corpora used in this study are specified in Section 2. The methods we adopted are shown in Section 3. The results are discussed in Section 4. Section 5 employs the results to protein name recognition. Finally, Section 6 concludes the remarks and lists some future works.

## 2. The Reference/Test Corpora

In the study, there are two important resources. One is the molecular dictionary in recognizing named entities in the text. The other one is the collection of the biological literatures.

The detection of protein/gene names presents

a challenging task because of their variant structural characteristics, their resemblance to regular noun phrases and their similarity with other kinds of biological substances. A rule-based method to recognize protein/gene names may propose more candidates than a dictionary-based one does, but the former may also introduce the false positive results that will provide the fault information about molecular entities and thus influence our further study. To avoid the complexity in the molecular entity identification, we focus on proteins only in the initial study. Furthermore, to get more accurate tagged corpus, we adopt the dictionary-based approach instead of the rule-based approach.

In the succeeding experiments, we employ a protein dictionary of size 2,227 entries. All the protein names in the dictionary are regarded as a protein class. From the view of the natural language processing, the context surrounding the protein class in the reference corpus shows important informative features. Protein/gene names those are not listed in the dictionary are called *unknown words* in natural language processing. The neighboring context of an unknown word is useful to disambiguate its function. If it is a protein or a gene, it often co-occurs with some particular keywords. The goal of this paper is to discover such keywords by collocation.

Medline is one of the headmost biomedical bibliographic database. In the study of mining the relationships between proteins, it is necessary to collect the literature about the proteins. The corpus for extracting biological keywords was downloaded from the PASTA website in Sheffield University [<http://www.dcs.shef.ac.uk/nlp/pasta>]. It includes 1,514 Medline abstracts on protein structures in period 1994-1998. In addition, to evaluate the effects of employing these keywords to protein name recognition, we need another test set that is exclusive of the PASTA corpus. The test corpus was downloaded from the Proteinhalt website in SICS, Sweden

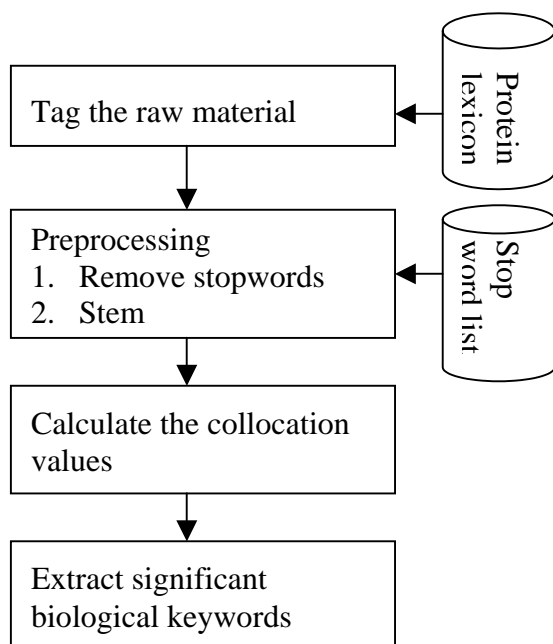
[<http://www.sics.se/humle/projects/prohalt>].

The test collection, which was used in Yapex project, contains 101 abstracts, including 48 Medline abstracts and 53 abstracts in GENIA corpus with partial modification [2].

In Medline text abstracts, different markers designate different kinds of information. They are TI (the title of the published paper), AU (the list of author names), NA (author addresses), JN (the journal references, including year, volume, number and page numbers), IS (the ISSN), AB (the text abstract), KP (keywords used by journals), WA (keywords used by authors) and PA (patterns). Since complete sentences contain more relationships between molecular entities, only TI and AB are used for investigation in this study.

### 3. Method

The overall architecture of our method is shown in Figure 1.



**Figure 1. Flow of Keyword Mining**

To extract biological keywords from scientific text, we require an informative corpus in which

protein names have been identified. Thus, we prepare a tagged biological corpus by looking up the protein lexicon in the first step. Then, in the preprocessing stage for the next analysis, two tasks are aimed as follows: Common stop words are removed and the stemming procedure is applied in order to gather and group more informative words. Next, the collocation values of protein names and their surrounding words are calculated. Finally, we use these values to tell which neighbouring words are the desired biological keywords. The detailed steps are specified thoroughly in the following subsections.

#### 3.1 Step 1: Tagging the Corpus

To calculate the collocation values of the designated words with proteins from the literature, it is necessary to recognize protein names at first. The task of identifying protein names is not easy since proteins have no consistent nomenclatures, and discovers or authors sometimes use many different spelling variations for the same molecular entities. This problem has been studied from two different perspectives. On the one hand, rule-based approaches take advantage of the morphological and part-of-speech information, as well as keywords to tag protein names. On the other hand, dictionary-based approaches make a partial or full pattern matching to dictionary entries. Usually, the rule-based approaches produce higher recall rate while dictionary-based approaches reach high precision performance. In our keyword extraction, a corpus with high precision is more important than high recall but with low precision. Thus, for the preparation of a tagged corpus, we adopted the dictionary-based approach, i.e., full pattern matching between the dictionary entries and the words in the corpus. The following shows a tagging example. A pair of tags `<NAME TYPE="PROTEIN"> ... </NAME>` is marked on the proteins.

*<NAME TYPE="PROTEIN"> Chloroperoxidase </NAME> (CPO) is a versatile heme-containing enzyme that exhibits <NAME TYPE="PROTEIN"> peroxidase </NAME> , <NAME TYPE="PROTEIN"> catalase </NAME> and <NAME TYPE="PROTEIN"> cytochrome P450 </NAME> -like activities in addition to catalyzing halogenation reactions.*

### 3.2 Step 2: Preprocessing

#### 3.2.1 Step 2.1: Exclusion of Stopwords

Stopwords are common English words (such as preposition “in” and article “the”) that frequently appear in the text but are not helpful in discriminating special classes. Because they are distributed largely in the corpus, they should be filtered out in this preprocessing step. The stopword list in this study was collected with reference to the stoplists of Fox [4], but the words also appearing in the protein lexicon are removed. For example, “of” is a constituent of the protein name “capsid of the lumazine”, so “of” is excluded from the final stoplist. In the experiments, 387 stopwords were used.

#### 3.2.2 Step 2.2: Stemming

Stemming is a procedure of transforming an inflected form to its root form. For example, “inhibited” and “inhibition” will be mapped into the root form “inhibit” after stemming. Because the reference corpus in the experiments is not large enough, stemming can group the same word semantics and reflect more information around the proteins.

### 3.3 Step 3: Computing Collocation Statistics

Collocations consist of two or more words and are characterized by limited compositionality. The collocation words with proteins specify that they often co-occur with protein names. In this

study, we calculate three collocation statistics to find the significant keywords about proteins.

#### Frequency

The collocations are selected by frequency. In order to gather more flexible relationships, here we define a collocation window that has five words on each side of protein names. And then collocation bigrams at a distance are captured. In general, more occurrences in the collocation windows are preferred, but the standard criteria for frequencies are not acknowledged. Hence, another collocation model is considered.

#### Mean and Variance

The mean value of collocations can indicate how far collocates are typically located from protein names. Furthermore, variance shows the deviation from the mean. If the standard deviation is equal to zero, it says that the collocates and the protein names always occur at exactly the same distance equal to the mean value. If the standard deviation is low, two words usually occur at about the same distance, i.e., near the mean value. If the standard deviation is high, then the collocates and the protein names occur at random.

We use the following formulas to calculate means and standard deviations, respectively.

$$\bar{d}_i = \frac{\sum_{j=1}^{n\_count_i} d_{ij}}{n\_count_i}$$

$$s_i = \sqrt{\frac{\sum_{j=1}^{n\_count_i} (d_{ij} - \bar{d}_i)^2}{n\_count_i - 1}}$$

Where  $\bar{d}_i$  is the average distance for word  $i$  in the collocation windows.  $d_{ij}$  is the distance of the  $j$ -th occurrence of word  $i$  away from proteins in the collocation windows. For example,  $d_{ij}=-1$  means the  $j$ -th occurrence of word  $i$  is located to the directly left of the

proteins in the collocation window.  $n\_count_i$  is the total occurrences of word  $i$ .  $s_i$  is the standard deviation of  $d_{ij}$ .

### *t*-test Model

When the values of mean and variance have been computed, it is necessary to know if two words do not co-occur by chance. Besides, we also have to know if the standard deviation is low enough. In other words, we have to set a threshold in the above approach. To get the statistical confidence that two words have a collocation relationship, *t*-test hypothesis testing is adopted.

The *t*-value for each word  $i$  is formulated as follows:

$$t_i = \frac{\bar{x}_i - u_i}{\sqrt{s_i^2 / N}}$$

Where

$$N = 4n - 15,$$

$$\bar{x}_i = \frac{n\_count_i}{N},$$

$$s_i^2 = p_i \times (1 - p_i),$$

$$p_i = n\_count_i / n,$$

$$u_i = p_{protein} \times p_i, \text{ and}$$

$p_{protein}$  is the probability of protein in the corpus.

When  $\alpha$  (confidence level) is equal to 0.005, the value of *t* is 2.576. In the *t*-test model, if the *t*-value is larger than 2.576, the word is regarded as a good collocate with 99.5% confidence.

### 3.4 Step 4: Extracting Collocation Keywords

The goal in this step is to extract good collocation keywords to facilitate the relationship discovery between genes, gene products or proteins. From the past papers on the extraction of the biological information, such as [1, 9, 11], verbs are the major targets.

This is because many of the subject and the object terms related to these verbs are names of genes or proteins. To assure that the collocation words selected in Step 3 are verbs, we assign parts of speech to the words. If one of the parts of speech of candidates in the lexicon is verb, then they are kept as collocation keywords.

## 4. Results and Discussions

Three kinds of statistical values were generated in Step 3. Frequency-based approach is not suitable. For example, both *bind* and *signal* are good collocates with proteins, but the frequencies of “bind” and “signal” are 365 and 9, respectively. A low threshold strategy will keep both of these two words, but many false candidates will pass the threshold at the same time. Hence, we cannot decide collocation keywords only by frequencies. For the similar reason, the mean-and-variance statistics cannot be solely used to extract keywords. For example, “phosphoinositide” owns zero standard deviation, but it only occurs twice in the corpus. Therefore, we consider the words selected by the *t*-test model.

### 4.1 Results of Extracting Keywords

Of the 4,782 different stemmed words appearing in the collocation windows, there are 541 collocations generated in Step 3. The collocation words are not tagged with parts of speech, so the output may contain nouns, prepositions, numbers, verbs, *etc.* The results from Step 3 with the highest 15 *t*-value are listed in Table 1.

In Table 1, the “Word” column lists the collocates found in Step 3. The “Freq” column shows the results about the frequency. The “Avg-Dist” column represents the average distance in the collocation windows. The “STD-Dev” column denotes the standard deviation. The “*t*-value” column describes the results in the *t*-test model. Under the “Word”

column, a special symbol <protein> maps all protein names listed in the corpus to the protein class we have defined earlier.

**Table 1. The Collocates with the Highest 15  $t$ -value in Step 3**

Word	Freq	Avg-Dist	STD-Dev	$t$ -value
the	3274	-0.743	3.298	53.455
of	2922	-1.081	2.808	50.493
and	1087	0.438	3.127	30.778
a	1016	0.754	3.203	29.755
structure	819	-1.591	2.999	26.713
<protein>	802	0.000	3.065	26.434
to	715	0.330	3.216	24.958
with	503	0.843	2.955	20.932
from	498	1.253	2.204	20.828
domain	387	-0.641	3.042	18.360
bind	365	-0.164	3.269	17.830
crystal	354	-2.689	2.638	17.559
complex	348	0.198	2.904	17.410
protein	289	0.453	3.420	15.865
an	226	0.942	3.199	14.030

After Step 4, there remained 154 collocation keywords with verbal part of speech, as shown in Appendix. Partial set of collocates with their related collocation values are illustrated in Table 2.

**Table 2. The Collocates with the Highest 15  $t$ -value in Step 4**

Word	Freq	Avg-Dist	STD-Dev	$t$ -value
structure	819	-1.591	2.999	26.713
bind	365	-0.164	3.269	17.830
complex	348	0.198	2.904	17.410
determine	193	1.233	3.079	12.965
activate	188	0.138	3.254	12.796
interact	116	-0.388	3.256	10.051
form	105	-0.467	3.308	9.562
inhibit	104	0.875	3.039	9.517
fold	102	0.049	3.442	9.425
contain	92	1.478	2.756	8.951
reveal	77	1.143	2.674	8.189
like	74	0.419	2.735	8.028
mutate	72	-0.778	3.311	7.918
sequence	70	-0.271	3.392	7.808
specify	67	0.507	3.616	7.638

Like Table 1, in “Word” column, it shows the useful verbal keywords those are usually accompanied with protein names. In “Freq” column, we know their occurrence times. In “Avg-Dist” and “STD-Dev” columns, it identifies the possible positions of keywords where they usually appear around the proteins. And the result is ordering according to “ $t$ -value” column.

The attractive enlightenment is observed from Table 2. For example, the verb “interact” is located at average “-0.388” position with the value of standard deviation “3.256”. The minus sign denotes that the word is on the left side of the protein. The absolute value of distance minus 1 indicates how many words are inserted in between the collocate and the protein class. The values of “STD-Dev” reflect that “interact” is usually situated in the collocation windows, but not at the exact position. The following two examples demonstrate that “interact” appears at two different locations. In the first example, “interact” is located at the right third word beside the protein *ubiquitin*; but in the second example, “interact” is beside the left second location of the protein name *ferredoxin* and the left fourth location of the protein *flavodoxin*. It shows why we cannot entirely depend on the “Avg-Dist” value, and the “STD-Dev” value must be also measured.

First example:

*As <NAME TYPE="PROTEIN"> ubiquitin </NAME>-conjugating enzymes interact with different substrates or other accessory proteins in the ubiquitination pathway, these variable surface regions may confer distinct specificity to individual enzymes.*

Second example:

*The model remain free to interact with <NAME TYPE="PEOTEIN"> ferredoxin </NAME> and <NAME TYPE="PROTEIN"> flavodoxin </NAME>, the physiological partners of <NAME TYPE="PROTEIN"> ferredoxin </NAME>: NADP(+) reductase.*

Now we have to estimate the performance of our method. The direct way is to ask an expert to examine the resultant keywords. However, different assessors may have inconsistent evaluation. Here we adopt an indirect approach. We verify the results from two different views. The first is to make sure if our keyword set contains those that most experts referred in their papers. The second is to apply the keyword set to certain application, e.g., protein name recognition, and to confirm if the performance of this application is improved when the keyword set is introduced. The first evaluation is listed in Table 3, and the second evaluation will be discussed in Section 5.

In Table 3, “A” (Author) is the surname of the first author in the references [1, 9, 11, 12, 15, 16, 17]. In these literatures, the authors use

verbs as keywords to support the interactions or pathways of proteins or genes. The number and the verbs listed in the literatures are listed in the “SV” (suggested\_verbs) column. The “F” (finding) column is the output in Step 4. The “U” (uncontained) column shows the suggested verbs that are not contained in the biological corpus. The “NF” (not\_found) column denotes the number and the verbs they are not considered as good keywords by our method. And the “P” column that denotes the performance is calculated as follows.

$$|finding| / (|suggested\_verbs| - |uncontained|)$$

In this formula, the uncontained words, which are absent from the corpus, is neglected in the performance evaluation.

**Table 3. First Evaluation Results in Step 4**

A	SV	F	U	NF	P
Blaschke	14 (acetylate, activate, associated with, bind, destabilize, inhibit, interact, is conjugated to, modulate, phosphorylate, regulate, stabilize, suppress, target)	10 (activate, bind, inhibit, interact, modulate, phosphorylate, regulate, stabilize, suppress, target)	2 (acetylate, is conjugated to)	2 (associated with, destabilize)	83.33 %
Ng	8 (inhibit, suppress, negatively regulate, activate, transactivate, induce, upregulate, positively regulate)	4 (inhibit, suppress, activate, induce)	3 (transactivate, upregulate, positively regulate)	1 (negatively regulate)	80%
Ono	4 (interact, associate, bind, complex)	4	0	0	100%
Park	12 (activate, accelerate, augment, induce, stimulate, require, up-regulate, inhibit, abolish, block, down-regulate, prevent)	5 (activate, induce, stimulate, require, inhibit)	2 (augment, down-regulate)	5 (accelerate, up-regulate, abolish, block, prevent)	50%
Sekimizu	7 (activate, bind, interact, regulate, encode, signal, function)	7	0	0	100%
Thomas	10 (activate, inhibit, modulate, suppress, isolate, promote, characterize, interact (with), associate (with), bind (to))	10	0	0	100%
Yakushiji	5 (bind, make (complex with), observe, incubate, induce)	3 (bind, observe, induce)	2 (make complex with, incubate)	0	100%

Total 30 keywords were cited in the papers [1, 9, 11, 12, 15, 16, 17]. Among these, 22 were extracted in Step 4. The average performance is 73.33%. The set of missing collocation words is {associated with, accelerate, abolish, block, destabilize, negatively regulate, prevent, up-regulate}. Six of these eight words come from pathway discovery except “associated with” and “destabilize”. Because our reference corpus is central on protein structures, i.e., deficiency of pathway information, the related keywords may not be extracted. On the other hand, compound words like “associated with” and “negatively regulate” cannot be found because the object collocations extracted by the previous algorithm are single words.

## 4.2 Results of Extracting Key Phrases

Compound words are possible key phrases from the above discussion. In the second experiment, we apply the early proposed method with window size of one word on each side of the original single keywords. In this way, the standard deviations of collocations play a more important role than those in the first round. After selection of standard deviations less than 0.4, “associated with” is discovered in this phase, but “negatively regulated” is still considered as a bad collocate in the  $t$ -test model. Consequently, the performance in the first row of Table 3 is improved to 91.67%. If the influence of pathways, i.e. Ng and Park [9, 12] used, is removed, the average performance is 95.45% (21/22). Otherwise, it is 76.67% (23/30).

Part of compound collocates with prepositions are shown in Table 4 and some compound collocates with nouns are represented in Table 5. There are 99 compound collocation words with 99.5% confidence and the values of standard deviations are less than 0.4. These results give more objective suggestion about compound collocation keywords.

**Table 4. Examples of Compound Collocates with Prepositions**

Word	Freq	STD-Dev	$t$ -value
associate with	71	0.333	8.423
bind by	10	0.000	3.127
bind near	8	0.000	2.797
complex at	7	0.000	2.633
complex between	10	0.000	3.147
complex to	38	0.453	6.135
complex with	445	0.134	21.008
effect of	83	0.309	9.105
interact with	163	0.000	12.762
isolated from	27	0.534	5.195

**Table 5. Examples of Compound Collocates with Nouns**

Word	Freq	STD-Dev	$t$ -value
gtpase activation	13	0.000	3.604
membrane association	11	0.000	3.315
binding sites	106	0.000	10.182
receptor bind	7	0.348	7.973
dna complex	50	0.000	7.037
complex structure	25	0.000	4.976
phage display	7	0.134	2.643
overhauser effect	25	0.000	4.997
genes encode	8	0.000	2.828
structure function	9	0.000	2.998

## 5. Protein Name Recognition

For protein name recognition, rule-based systems and dictionary-based systems are usually complementary. Rule-based systems can recognize those protein names not listed in a dictionary, but some false entities may also pass at the same time. Dictionary-based systems can recognize those proteins in a dictionary, but the coverage is its major drawback. In this section, we will employ the keywords mined earlier to help identify the molecular entities. Recall that the keywords are good collocates with protein names. The protein name recognizer Yapex from Olsson, *et al.* [10], which is a ruled-based system, was adopted. After tagging protein names on a corpus by

Yapex system, the keyword set is served as restrictions to filter out wrong protein names.

The following filtering strategies are proposed. Assume M0 is the output generated by Yapex.

- M1: For each candidate in M0, check if one of the keywords is found in its collocation window. If yes, tag the candidate as a protein name. Otherwise, discard it.
- M2: Some of the keywords may be substrings of protein names. We relax the restriction in M1 as follows. If one of the keywords appears in the candidate or in the collocation window of the candidate, then tag the candidate as a protein name; otherwise, discard it.
- M3: Some protein names may appear more than once in a document. They may not always co-occur with one of the keywords in each occurrence. In another word, the protein candidate and the keyword may co-occur in the first occurrence, the second occurrence, or even the last occurrence. We revise M1 and M2 as follows to capture this phenomenon. During checking if there exists one of the keywords co-occurring with a protein candidate, the candidate without any collocate is kept undecidable instead of definite no. After all the protein names are examined, those undecidable candidates may be considered as protein names when one of their occurrences containing any collocate. In other words, as long as a candidate has been confirmed once, it is assumed to be a protein throughout. In this way, there are two filtering alternatives M31 and M32 from M1 and M2, respectively.

To get more objective evaluation, we utilized another corpus of 101 abstracts described in Section 2. Using the test corpus and answer keys supported in Yapex project, the evaluation results are listed in Table 6.

**Table 6. Second Evaluation Results**

	Precision	Recall	F-score
M0	70.90%	69.53%	70.22%
M1	79.18%	56.10%	67.64%
M2	79.29%	56.66%	67.98%
M31	81.97%	66.84%	74.41%
M32	81.94%	67.14%	74.54%

Compared to the baseline model M0, the precision rates of all the four models using collocation keywords were improved more than 8%. The recall rates of M1 and M2 decreased about 13%. Thus, the overall F-scores of M1 and M2 decreased about 2% compared to M0. In contrast, if the decision of tagging was deferred until all the information were considered, then the recall rate decreased only 2% and the overall F-scores of M31 and M32 increased 4% relative to M0. The best one, M32, improved the precision rate from 70.90% to 81.94%, and the F-score from 70.22% to 74.54%. That meets our expectation, i.e., to enhance the precision rate, but not to reduce the significant recall rate. It also indicates that the keywords mined in the earlier section are very useful in protein name recognition.

## 6. Concluding Remarks

We have shown a fully automatic way of mining biological keywords from scientific text in the protein domain. The methods were based on collocation statistics, and the performance reached to 95.45% if we focused on the interaction between proteins. Besides, we have also shown the extracted keywords are useful in enhancing the precision rate of protein name recognition. In the application of protein name recognition, it enhances the precision from 70.90% to 81.94% on Yapex system. The same approach can be extended to other domains like gene, DNA, RNA, drugs, and so on. Similarly, the keywords for pathway discovery can be extracted when the reference corpus is enlarged to cover the relevant scientific documents.

Some improvements and future work may proceed. In the first place, a larger text corpus should be tested owing that the larger one contains more complete information. As mentioned in Table 3, uncontained words such as “incubate” might appear in the larger corpus and the collocation model can be applied to test if they are good keywords. Better quality of keywords will improve the performance of named entity recognition in Section 5. Secondly, including the abstracts about pathways can elicit good collocation words on pathways. In the subsequent development, the incorporation with papers on pathways will make improvement to the first performance expected at least 76.67%. Thirdly, a larger background protein dictionary should replace the one in our experiment that is derived from PDB, SCOP, CATH and SWISS-PROT databases. In such a way, more information in the collocation windows can be gathered and helps to retrieve more complete collocation keywords. Fourthly, we will employ the keywords or key phrases mined in this paper to find interaction between proteins or genes, as some researchers [1, 11, 14, 15, 16] have adopted. Finally, how to improve recall rate and precision rate of the protein name tagged corpus is another considerable direction.

In summary, the work presented herein represents significant evidence toward mining relationships among textual sources of biological knowledge, e.g., protein-protein, protein-gene, drug-gene, drug-disease, and so on. Lots of applications, such as name, interaction and pathway extraction, will get benefit from this study.

### **Acknowledgements**

Part of research results was supported by National Science Council under the contract NSC-91-2213-E-002-008. We also thank Dr. George Demetriou in the Department of the Computer Science of the University of Sheffield, who kindly supported the resources in this work.

### **Appendix**

act, activate, adopt, affect, allow, analyse, appear, arrange, assemble, associate, base, belong, bind, bond, bridge, calculate, call, carry, catalyze, cause, center/centre, change, characterize, charge, class, cleave, close, coil, compare, complex, composed, comprise, conclude, conserve, consist, constitute, contact, contain, coordinate, correlate, correspond, crystallize, cycle, define, demonstrate, depend, derive, describe, design, detail, determine, differ, diffract, digest, dimerize, direct, discuss, display, disrupt, effect, encode, enhance, exhibit, exist, explain, express, extend, facilitate, find, fold, form, function, groove, hydrolyze, identify, implicate, inactive, include, indicate, induce, inhibit, initiate, insert, interact, involve, isolate, lack, lead, ligand, like, link, locate, loop, mediate, model, module, mutate, observe, obtain, occupy, occur, organize, oxidize, phosphorylate, play, position, predict, present, produce, promote, propose, protonate, provide, purify, react, recognize, reduce, refine, regulate, relate, repeat, replace, report, represent, require, resemble, resolve, result, reveal, select, sequence, serve, shape, share, show, signal, solve, stabilize, stimulate, strain, strand, structure, study, substitute, substrate, suggest, support, switch, synthesize, target, transfer, transport, understand, use.

### **References**

- [1] Blaschke, C., Andrade, M.A., Ouzounis, C. and Valencia, A. (1999) “Automatic Extraction of Biological Information from Scientific Text: Protein-Protein Interactions”, *Intelligent Systems for Molecular Biology 1999*, Heidelberg, Germany - AAAI Press, pp. 60-67.
- [2] Collier, N., Park, H. S., Ogata, N., Tateishi, Y., Nobata, C. and Ohta, T. (1999) “The GENIA project: Corpus-based Knowledge Acquisition and Information Extraction from Genome Research Papers”, *Proceedings of the*

*Annual Meeting of the European Chapter of the Association for Computational Linguistics*, June, 1999.

[3] Collier, N., Nobata, C. and Tsujii J.I. (2000) "Extracting the Names of Genes and Gene Products with a Hidden Markov Model", *Proceedings of 18<sup>th</sup> International Conference on Computational Linguistics*, pp. 201-207.

[4] Fox, C. (1992) *Lexical Analysis and Stoplists*. In *Information Retrieval: Data Structures and Algorithms*, Frakes, W. B. and Baeza-Yates, R., ed., Prentice Hall, pp. 102-130.

[5] Friedman, C., Kra, P., Yu, H., Krauthammer, M. and Rzhetsky, A. (2001) "BENIES: A Natural-language Processing System for the Extraction of Molecular Pathways from Journal Articles", *Bioinformatics*, 17, Suppl. 1, pp. S74-S82.

[6] Fukuda, K., Tsunoda, T., Tamura, A. and Takagi, T. (1998) "Toward Information Extraction: Identifying Protein Names From Biological Papers", *Proceedings of the Pacific Symposium on Biocomputing*, pp. 707-718.

[7] Hirschman, L., Park, J.C., Tsujii, J., Wong, L. and Wu, C.H. (2002) "Accomplishments and Challenges in Literature Data Mining for Biology", *Bioinformatics*, 18, pp. 1553-1561.

[8] Krauthammer, M., Rzhetsky, A., Morozov, P. and Friedman, C. (2000) "Using BLAST for Identifying Gene and Protein Names in Journal Articles", *Gene*, 259, pp. 245-252.

[9] Ng, S.K. and Wong, M. (1999) "Toward Routine Automatic Pathway Discovery from On-line Scientific Text Abstracts", *The 10<sup>th</sup> Conference on Genome Informatics*, 10, pp. 104-112.

[10] Olsson, F., Eriksson, G., Franzen, K., Asker, L. and Liden P. (2002) "Notions of Correctness when Evaluating Protein Name

Taggers", *The 19<sup>th</sup> International Conference on Computational Linguistics*, pp.765-771.

[11] Ono, T., Hishigaki, H., Tanigami, A. and Takagi, T. (2001) "Automated Extraction of Information on Protein-Protein Interactions from the Biological Literature", *Bioinformatics*, 17(2), pp. 155-161.

[12] Park, J.C., Kim, H.S. and Kim, J.J. (2001) "Bidirectional Incremental Parsing for Automatic Pathway Identification with Combinatory Categorical Grammar", *Proceedings of the Pacific Symposium on Biocomputing*, 6, pp. 396-407.

[13] Proux, D., Rechenmann, F. and Julliard, L. (1998) "Detecting Gene Symbols and Names in Biological Texts: A First Step Toward Pertinent Information Extraction", *The 9<sup>th</sup> Conference on Genome Informatics*, pp. 72-80.

[14] Proux, D., Rechenmann, F. and Julliard, L. (2000) "A Pragmatic Information Extraction Strategy for Gathering Data on Genetic Interactions", *Proceedings of the 8<sup>th</sup> International Conference on Intelligent Systems for Molecular Biology*, pp. 279-285.

[15] Sekimizu, T., Park, H.S. and Tsujii, J. (1998) "Identifying the Interaction Between Genes and Genes Products Based on Frequently Seen Verbs in Medline Abstract", *The 9<sup>th</sup> Conference on Genome Informatics*, pp. 62-71.

[16] Thomas, J., Milward, D., Ouzounis, C., Pulman, S. and Carroll, M. (2000) "Automated Extraction of Protein Interactions from Scientific Abstracts", *Proceedings of the Pacific Symposium on Biocomputing*, 4-9 January 2000, Honolulu, Hawaii, 5, pp. 538-549.

[17] Yakushiji, A., Tateisi, Y. and Miyao, Y. (2001) "Event Extraction from Biomedical Papers Using a Full Parser", *Proceedings of the Pacific Symposium on Biocomputing*, 6, pp. 408-419.