Two Experiments in Biological Term Annotation using Classification Methods

Sittichai Jiampojamarn, Vlado Kešelj and Nick Cercone
{*jiampoja, vlado, nick*}*@cs.dal.ca*

Faculty of Computer Science
Dalhousie University
Halifax, NS, Canada B3H 1W5

Abstract

The number of publications in biological research is growing rapidly. A large part of our research has been involved in making this biological research more accessible. Most of these publications are available online in an unstructured textual format, such as in the PubMeds MedLine web site. Reading every available article is a time-consuming task; an automatic method of extracting information from them is desirable. The first task in discovering knowledge from these publications is to identify biological terms in articles. This problem is challenging, involving issues such as tackling unknown words, long multi-word terms, and variant author styles to express biological terms.

We present two methods that extend classification algorithms to annotate biological terms. In the first experiment, we exploit special lexical and morphological characteristics of biological terms using different classification algorithms. The C4.5 classification algorithm showed outstanding performance. In the second experiment, we performed C4.5 classification to learn characteristic patterns of biological terms. We obtained 0.76 F-score in extraction of biological terms from the GENIA 3.02 corpus, which includes 2,000 paper abstracts.

Keywords *Information extraction, Bioinformatics, Text mining*

1. Introduction

Exciting new research into facets of biology had led to a tremendous surge in the amount of biological literature published. Most of these publications are available online such as in the PubMed's MedLine database [1]. Providing software support for knowledge discovery from the available publications in the biomedical area is a challenging endeavor. Knowledge discovery from online biological publications is difficult because the articles are typically unstructured text. The first step in making these articles more accessible is to annotate the biomedical terms, pointing out the interesting words in the documents, which are the most salient content-bearing words. After this step, one can proceed with detection of relations between the terms, pathways discovery from the literature. However, reading all articles and annotating manually by human experts becomes an infeasible task due to the large size of data. Automatically annotating the biological terms in these unstructured articles becomes a crucial task. The following is an example shown input text [2] and output text where biological terms are annotated in documents :

> Activation of the CD28 surface receptor provides a major costimulatory signal for T cell activation resulting in enhanced production of interleukin-2 (IL-2) and cell proliferation.
>
> Activation of the <term>**CD28 surface receptor**</term> provides a major costimulatory signal for <term>**T cell activation**</term> resulting in enhanced production of <term>**interleukin-2**</term> (<term>**IL-2**</term>) and cell proliferation.

Machine-learning methods are used to automatically annotate biological terms, such as gene names, proteins, organs, diseases, etc.—i.e., a list of terms that is specified by domain experts. This task is similar to the named-entity recognition task in the Message Understanding Conference (MUC). [1] Difficulties in identifying these terms are included on unseen terms, long multi-word terms, and various writing styles by different authors.

Biological terms can be single-word terms (e.g., Adenovirus, E1A, tublin, GATA-1) or multi-word terms (e.g., mouse interleukin-1 receptor alpha gene, large granular lymphocytes). Many multi-word terms can be written in different styles depending on the author's preferences. For example, some authors prefer abbreviated short terms, and

[1] The proceedings of the seven MUC conferences were published by the Morgan Kaufmann Publishers in 1990's. The Web site is http://www.itl.nist.gov/iaui/894.02/related_projects /muc/.

other prefer the full long form; e.g., “large granular lymphocytes” can be re-written as “LGL,” or “UDG” as “uracil-DNA glycosylase”.

To solve this problem, we used classification algorithms to identify biological terms by learning regular characteristics of terms rather than manual setting patterns or looking-up any biological dictionary. Our approach is based on relaxing fixed rules and generalizing the system to extract biological terms in different styles of research publications. We take advantage of special character-based biological term characteristics, including uppercase letters, digits, and symbols, as well as the biological concept words such as “gene,” “cell,” “protein”, and so on. Part-of-speech tagger [3] is also applied in automatic classification methods to learn the patterns of these special characteristics in biological terms.

The paper is organized as follows. In the next section we introduce some related work. In Section 3, two experiments in the problem of biological term annotation are described, and the results are presented and discussed. The conclusion and future work are presented in Section 4.

2 Related Work

Text mining in biology has been interested in many research groups and conferences, including the TREC genomics track [4] which has a goal to provide information support tools for genomic domain. A biological term annotation task plays an important role in measuring biologically meaningful shown in BioCreAtIvE [5]. Recently there have been published papers on the problem of biological term annotation and many of them made use of the GENIA corpus [2]. The current methodology in biological terms extraction task can be divided in two main approaches, rule-based methods and learning methods.

K. Fukuda et al. [6] focused on identifying protein names based on traditional rule-based methods in which the rules are manually set by observing some special characteristics in protein names such as uppercase letters, lowercase letters, digits, special symbols and characteristic words such as kinase, receptor, protein, etc. Then, they combine those “core terms” together as they describe that most protein names are compound multi-word terms. They obtain 0.94 precision and 0.98 recall on 50 research abstracts retrieved from MedLine.

N. Collier et al. [7] used HMM (Hidden Markov Model) to extract protein and DNA names in a small corpus of 100 MedLine abstracts. They defined the word features based on characteristics of known terms in training data set such as digit numbers, uppercase letters, lowercase letters, Greek letters, combination of uppercase, lowercase letters and digits, and special symbols. They achieved 0.73 F-score on a cross-validation test.

L. Venkata Subramaniam et al. [8] proposed Bioannotator, a biological terms annotation system which combines both a rule-based and a dictionary-based engine. They used a shallow parser to identify noun phrases in the documents, then each noun phrase is labeled whether it is biological term or not based on three dictionaries: Unified Medical Language Systems (UMLS)[9], LocusLink [10], and GeneAlias, as well as the rule engine. The rule engine is a set of regular expressions to recognize a word which contains uppercase letters, digits, special symbols, Greek letters, and characteristic words in biological terms such as amylase, cell, gene, amino, etc. The system achieved 0.64 F-score in exact matching and 0.90 F-score in partial matching with the answers in GENIA 1.1 corpus [2], which contains 670 research abstracts. This result showed that the boundaries of biological terms are difficult to detect. Although most of biological terms are nouns, these terms are usually just parts of proper noun phrases.

Our approach relies on the fact that biological terms usually have some special character-based characteristics, including uppercase letters, digits, and symbols, as well as the biological concept words such as “gene,” “cell,” “protein,” etc. Instead of manually setting rules, we use classification methods to learn the patterns of these special characteristics in biological terms.

3 Methods, experiments, results

We used existing classification methods to identify the terms automatically rather than to set up rules manually to capture some specific characteristics of biological terms. A classifier extracts boundaries of biological terms, starting and ending positions, from the plain text in documents. We used word n-grams to chunk out each sentence and extract feature attributes for training classifiers. The feature attributes are general characteristics of each word in a n-gram. However, this is still a question how to define the significant feature attributes, which capture distinction between biological terms from normal words. We set up two experiments to investigate the proper way to use the classification scheme for biological terms extraction.

3.1 First experiment

It has been shown in [6, 7, 8] that biological terms have some regular characteristics expressed in appearance of the uppercase letters with special symbols or digit numbers. Many terms contain Greek letters, expressed as English words, or have prefixes or suffixes of biological concepts (e.g., ase, cyt). The multi-word terms usually contain characteristic words of the biological domain such as cell(s), protein(s), amino, and gene(s). Therefore, we first set feature attributes to be :

- numbers of uppercase letters.
- numbers of digits.
- numbers of symbols.
- numbers of Greek letters shown in table 1.
- indicator of whether a word has a specific biological prefix or suffix concept.
- part-of-speech tag information tagged by bi-gram hidden Markov model POS tagger [3].

Alpha	Beta	Gamma	Delta	Epsilon
Zeta	Eta	Theta	Iota	Kappa
Lambda	Mu	Nu	Xi	Omicron
Pi	Rho	Sigma	Tau	Upsilon
Phi	Chi	Psi	Omega	

Table 1: Greek Letters

We use a sliding window of n words, i.e., a word n-gram model, that starts from the beginning of each sentence shown in table 2 and extracts feature attributes described above for each word. Additionally, with each word position we associate a class attribute that labels the position as a starting or ending position of a biological term, or as an undistinguished position. Classifiers are trained and evaluated on the task of detecting starting and ending positions of a term. This position information is further used in annotating biological terms.

Input sentence :

> The CD4 coreceptor interacts with non-polymorphic regions of major histocompatibility complex class II molecules on antigen-presenting cells and contributes to T cell activation.

Word n-gram :

The
The CD4
The CD4 coreceptor
CD4 coreceptor interacts
coreceptor interacts with
⋮
T cell activation.
cell activation.
activation.

Table 2: word n-gram example, where $n = 3$

We trained and tested our method on the GENIA 1.1 corpus [2], which contains 670 research abstracts with biological terms annotated by human experts. We used classification algorithms in the WEKA 3.4 machine learning tool [11] to perform our experiments. The result of extracting the starting positions in different classification methods are shown in table 3. Precision, Recall and F-score are common standards to evaluate the performance of a classifier defined as in equations 1, 2, and 3, where TP (true positive) is the number of terms which are correctly classified to the class, FP (false positive) is the number of terms which are correctly unclassified to the class, and FN (false negative) is the number of terms which are incorrectly unclassified to the class.

$$\mathit{Precision} = \frac{\mathit{TP}}{\mathit{TP} + \mathit{FP}} \tag{1}$$

$$\mathit{Recall} = \frac{\mathit{TP}}{\mathit{TP} + \mathit{FN}} \tag{2}$$

$$\mathit{F\text{-}score} = \frac{2 \times \mathit{Precision} \times \mathit{Recall}}{\mathit{Precision} + \mathit{Recall}} \tag{3}$$

As shown in table 3, we obtain the best performance from the C4.5 classification algorithm achieving the F-score of 0.64. In the same way, we obtained 0.72 F-score for the ending position accuracy. We combined both information from the classifier to extract biological terms based on starting and ending positions. The resulting performance decreased in accuracy to be 0.52 F-score for exact matching against the answers in the corpus shown in table 4. We explain this effect as an accumulated error from the starting-position and ending-position classification. Either an error in starting or ending position will lead us to a wrong term in the sense of exact matching. If the errors in determining the starting and ending positions were completely independent, we would have a probability of $0.71 \cdot 0.76 = 0.54$ of getting an exact match. The exact matching precision 0.65 is somewhat higher, which implies that detection of starting and ending positions are somewhat correlated. With respect to recall, the independence assumption implies an exact matching recall of $0.58 \cdot 0.69 = 0.40$, which is very close to actual 0.43. This experimental result justifies our conclusion that the decreased performance on exact matching is largely due to independent treatment of starting and ending position detection.

3.2 Second experiment

In section 3.1, the experiment revealed us some problems in automatically detecting boundaries in multi-word biological terms. Tracking only starting and ending position of each term does not provide us with enough information to annotate terms in overall, dealing with both long words and single words at the same time.

Classifiers	Precision	Recall	F-score
NaiveBayes	0.54	0.46	0.50
C4.5	0.71	0.58	0.64
AdaBoostM1	0.67	0.25	0.37
LogisRegress	0.66	0.45	0.53
SMO	0.69	0.34	0.46
IB1	0.69	0.58	0.63
Rule Part	0.72	0.54	0.62

Table 3: Classifications performance

Annotate	Precision	Recall	F-score
Starting	0.71	0.58	0.64
Ending	0.76	0.69	0.72
Exact match	0.65	0.43	0.52

Table 4: Annotation performance

Target Class	Precision	Recall	F-score
Starting	0.58	0.44	0.50
Middle	0.68	0.26	0.38
Ending	0.69	0.49	0.57
Single	0.60	0.61	0.60
Relevant	0.90	0.65	0.76

Table 5: Word tag positions performance

In the second experiment, we added more target classes for the classifier. Unlike earlier, we classify each instance in five classes as starting, middle, ending, single, and non-relevant. The starting and ending classes indicate the beginning and ending positions of biological terms while the middle class indicates the words between starting and ending positions in a multi-word biological term. The single class means that those biological terms are only one word long. Finally, the other words, which are not biological terms, will be classified as the non-relevant class.

From the first experiment we know that biological terms are correlated with the occurrences of uppercase letters, digits, and special symbols. These features show the difference between general words and biological words as the higher number of uppercase and symbol letters. However, they do not cover the case where short biological words have a small number of uppercase letters or symbols. The words with uppercase letters in the middle have higher possibility to be biology related than uppercase letters at the beginning, which can be confused at the beginning of sentences and in proper nouns. From our observation, the number is not as important as pattern of character features formed in a word.

We further refined the sensitivity of the classification algorithms by extracting word feature patterns, including Greek letters, Leading words, Uppercase, Lowercase, Digits, Hyphen, Plus, Slash, OpenParen, CloseParen, OpenSquare, CloseSquare, Percent, and other symbols. The word feature patterns provide the capability to annotate unseen biological words which have the same pattern even they are not in the training set. To reduce dimensionality, we limit the number of features recorded for each word in an n-gram window to m. For example for $m = 4$, the term “CD28-mediated” generates the pattern as Uppercase, Digits, Hyphen, and Lowercase. In case a word has more features than the value m, only the first m features are extracted and the rest of the features are ignored. In another case, if a word has less features than the value m, the dummy feature “none” is appended up to the m value. In this case, the number of feature attributes for each instance in the classifier is $m \times n$, where m is the value of word features we extract from n-word n-grams.

We used the GENIA 3.02 corpus which contains 2,000 paper abstracts. As the result in table 5, the classifier identified single biological terms with higher accuracy than multi-word terms indicated by lower accuracy on starting, middle and ending classes. Hence, there is not much difference or unique characteristics among starting words, ending words, and the words in the middle. However, when we combined all classes above as one class labeled “relevant,” the classifier provided a higher accuracy on making the decision whether the word is the biological term or a normal word with precision 0.90, recall 0.65 and F-score 0.76.

4 Conclusions

In this paper, we presented two experiments for biological terms annotation using classification methods. The classifiers captured the regular characteristics of biological terms from training data, then they were used to detect whether the terms are biology related or not. We used different classification algorithms. The experiments demonstrated that the C4.5 algorithm is the suitable classification method for annotation of biological terms. We used general word features such as uppercase and lowercase letters, digits, special symbols, as feature attributes for the classifier to learn biology-related term patterns. These features are general and can be adapted for other domains. We got 0.76 F-score on distinction between biological and normal terms. However, extracting exact multi-word terms remains to be improved in future.

Our results are comparable with dictionaries and rule-based systems [8] while we reduced the manual effort in creating rules and patterns and tested the results on a larger corpus. However, the selection of feature attributes used in classification need to be improved for a better performance, especially in exact annotation of the boundaries of multi-word biological terms.

In future work, we plan to enrich feature attributes with new general features relevant to this problem as well as consider rough set theory [12] helping in feature selection for classification [13]. The problem of defining boundaries of each term can be addressed by considering syntactic analysis.

References

[1] MedLine. http://www.ncbi.nlm.nih.gov/PubMed/, Accessed June,2004.

[2] J.Tsujii *et al.* Genia corpus. http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/, Accessed June, 2004.

[3] A.Coburn M.Ceglowski. Lingua::EN::Tagger Part-of-speech tagger for English natural language processing. http://theoryx5.uwinnipeg.ca/CPAN/data/Lingua-EN-Tagger/Tagger.html, Accessed Nov, 2003.

[4] Text Retrieval Conference (TREC) Genomics Track. http://medir.ohsu.edu/~genomics/, Accessed Sept,2004.

[5] BioCreAtIvE Critical Assessment of Information Extraction systems in Biology. http://www.pdg.cnb.uam.es/BioLINK/BioCreative.eval.html, Accessed Sept,2004.

[6] K. Fukuda, T. Tsunoda, A. Tamura, and T. Takagi. Toward information extraction: Identifying protein names from biological papers. In *Proc. of the Pacific Symp. on Biocomputing*, pages 707–718, 1998.

[7] N. Collier, C. Nobata, and J. Tsujii. Extracting the names of genes and products with a hidden Markov model. In *Proc. of COLING'2000, Saarbrucken*, pages 201–207, 2000.

[8] L. Venkata Subramaniam, Sougata Mukherjea, Pankaj Kankar, Biplav Srivastava, Vishal S.Batra, Pasumarti V. Kamesam, and Ravi Kothari. Information extraction from biomedical literature: Methodology, evaluation and an application. In *Proc. of the CIKM'12*, pages 410–417, 2003.

[9] UMLS. http://umlsks.nlm.nih.gov, Accessed June, 2004.

[10] Pruitt K.D. and Maglott D.R. Locuslink. http://www.ncbi.nlm.nih.gov/locuslink/, Accessed June, 2004.

[11] Weka 3: Machine learning software in Java. http://www.cs.waikato.ac.nz/ml/weka/, Accessed Dec, 2003.

[12] Z. Pawlak. Rough sets: Theoretical aspect of reasoning about data. In *Kluwer Academic Publishers*, 1991.

[13] H. Itakura T. Wakaki and M. Tamura. Rough set-aided feature selection for automatic web-page classification. In *Proc. of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence (WI'04)*, 2004.