Result Identification for Biomedical Abstracts Using Conditional Random Fields

Ryan T.K. Lin[1], Hong-Jei Dai[13], Yue-Yang Bow[1], Min-Yuh Day[1], Richard Tzong-Han Tsai[2], and Wen-Lian Hsu[13].
[1]Institute of Information Science, Academia Sinica, Taipei, Taiwan, R.O.C.
[2]Dept. of Computer Science & Engineering, Yuan Ze Univ., Taoyuan, Taiwan, R.O.C.
[3]Dept. of Computer Science, National Tsing-Hua Univ., Hsinchu, Taiwan, R.O.C.

Abstract

For biomedical research, the most important parts of an abstract are the result and conclusion sections. Some journals divide an abstract into several sections so that readers can easily identify those parts, but others do not. We propose a method that can automatically identify the result and conclusion sections of any biomedical abstracts by formulating this identification problem as a sequence labeling task. Three feature sets (Position, Named Entity, and Word Frequency) are employed with Conditional Random Fields (CRFs) as the underlying machine learning model. Experimental results show that the combination of our proposed feature sets can achieve F-measure, precision, and recall scores of 92.50%, 95.32% and 89.85%, respectively.

1. Introduction

The number of biomedical publications now available to researchers is overwhelming. In recent years, there has been a great deal of activity in the field of biomedical text mining, and a range of text-mining applications have been developed to facilitate research conducted by biologists and database curators [1-3]. These tools perform functions like recognizing named entities or identifying relationships between entities. Generally, such applications analyze the entire abstract, without distinguishing the introduction from the method, the result or the conclusion. However, in the biomedical field the result and conclusion sections of an abstract usually describe the true contribution of a paper. Therefore, it would be advantageous to distinguish them from the other parts of the abstract so they can be extracted for further text mining, and thereby help researchers to quickly focus on new findings and the contributions presented.

Since most journal formats do segment abstracts, we must develop a method to split abstracts into different sections based on linguistic features. In this work, we introduce a machine learning (ML) based method to identify the result or the conclusion section automatically. For convenience, we will refer to the result and conclusion sections as the "result section". We have chosen to use the Conditional Random Fields (CRFs) [4] ML model because it relaxes independence assumption and it performs better than other ML models [5]. We also propose three feature sets, position, named entity and word frequency, which can effectively identify whether sentences belong to the result section or not.

To evaluate our proposed methods, we selected the training and test data from a controlled source namely hypertension-gene relation articles. Biologists with many years experience in hypertension research helped us annotate the controlled articles with the result section boundary. We conducted seven experiments on this corpus to evaluate the proposed feature sets and the combinations of them.

The remainder of this paper is organized as follows: Section 2 contains a review of related works. In Section 3, we describe the CRFs and our proposed feature set. Section 4 reports on the experiment results. In Section 5, we discuss why Position + NE feature is better than Position, NE, or WF feature individually and explain how these features work. Then, in Section 6, we summarize our conclusions.

2. Related Work

Several result identification approaches have been proposed. For example, Ruch et al. [6] used the Bayesian classifier with word and position feature, which achieved an F-score of 85% in identifying the conclusion section of abstracts.

Lin et al. and Wu et al. [7, 8] proposed using the generative model, namely a hidden markov model (HMM), to analyze structural abstracts. Lin et al. [7] used generated bigram language models for each section using Kneser-Ney discounting and Katz backoff. Their system achieved an F-score of 89.8% for result section and 89.7% for conclusion section. Wu et al. [8] proposed a six steps of learning process to train a collocation classifier and achieved 80.54% precision.

Yamamoto et al. [9] developed a system to classify sentences in abstracts into sections. They trained a linear-

978-1-4244-2660-7/08/$25.00 ©2008 IEEE

Support Vector Machine (SVM) classifier with features such as unigram, subject-verb, verb tense, relative sentence location, and sentence score (i.e., average TF-IDF score of constituent words). Their method achieved F-Score of 87.2% and 89.8% for result and conclusion, respectively.

Shimbo et al. [10] used SVM to classify sentences represented by words, word bigram, and contextual information and reported 91.9% accuracy.

3. Method

3.1. Formulation

In the result identification problem, we regard each sentence in an abstract as a token. Each token is associated with a tag that indicates whether or not it belongs to the result section and its location within the result section, that is, *B-RS*, *I-RS*, and *O*. The first two tags denote, respectively, the beginning token and the following token of the result section; the last tag indicates that a token is not part of the result section. This problem can then be formulated as the problem of assigning one of three tags to each token.

3.2. Conditional Random Fields

3.2.1. The Model

CRFs are undirected graphical models trained to maximize a conditional probability [11]. A linear-chain CRF with parameters $\Lambda=\{\lambda_1, \lambda_2, \ldots\}$ defines a conditional probability for a state sequence $\mathbf{y} = y_1 \ldots y_T$ given an input sequence $\mathbf{x} = x_1 \ldots x_T$ to be

$$P_\Lambda(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z_\mathbf{x}} \exp\left(\sum_{t=1}^{T} \sum_{k} \lambda_k f_k(y_{t-1}, y_t, \mathbf{x}, t) \right), \quad (1)$$

where Z_x is the normalization that makes the probability of all state sequences sum to one; $f_k(y_{t-1}, y_t, \mathbf{x}, t)$ is often a binary-valued feature function and λ_k is its weight. The feature functions can measure any aspect of a state transition, $y_{t-1} \rightarrow y_t$, and the entire observation sequence, x, centered at the current time step, *t*. For example, one feature function might have value 1 when y_{t-1} is the state *B-RC*, y_t is the state *I-RC*, and x_t is 1st position. Large positive values for λ_k indicate a preference for such an event; large negative values make the event unlikely.

The most probable label sequence for **x**,

$$\mathbf{y}^* = \underset{y}{\operatorname{argmax}}\ P_\Lambda(\mathbf{y} \mid \mathbf{x}),$$

can be efficiently determined using the Viterbi algorithm [12].

The parameters can be estimated by maximizing the conditional probability of a set of label sequences, each given their corresponding input sequences. The log-likelihood of a training set $\{(x_i, y_i): i = 1, \ldots, M\}$ is written as:

$$L_\Lambda = \sum_i \log P_\Lambda(\mathbf{y}_i \mid \mathbf{x}_i)$$

$$= \sum_i \left(\sum_{t=1}^{T} \sum_{k} \lambda_k f_k(y_{t-1}, y_t, \mathbf{x}, t) - \log Z_{\mathbf{x}_i} \right).$$

To optimize the parameters in CRFs, we uses a quasi-Newton gradient-climber BFGS [13]

3.2.2. Feature Set

In this section, we now describe the features used in the proposed approach, namely position, named entity (NE), and word frequency (WF).

Position Feature

The relative position of a sentence in an abstract is an important feature because the result section is often described at the end of the abstract. Therefore we have designed a position feature to represent the sentence's position, which can be calculated by the following equation:

$$f_{\text{Position}}(s, S) = \left\lceil \frac{normalized\ factor \times s}{S} \right\rceil,$$

where *s* is the sentence's position in the abstract, and *S* is the total numbers of sentences in an abstract. In this work, we use a normalization factor of 10 to adjust range of the feature value. By taking ceiling, we discriminate the value of f_{Position} into an integer ranging between 1 and 10.

NE Feature

Since the title can be treated as the summary of an abstract, it may contribute some information related to the result passage. Named entities (NE) of the title could provide such information. In our work, we first employ our NE recognizer, NERBio [5], to extract NEs in the sentence of the title and the current sentence. Then, two sub-features are designed to represent how many NEs are shared in these two sentences. The first is f_{bNE}, which is a binary feature defined as follows:

$$f_{bNE}(\{NE_s\}, \{NE_{title}\}) = \begin{cases} 1, & \{NE_s\} \cap \{NE_{title}\} = \emptyset \\ 0, & \{NE_s\} \cap \{NE_{title}\} = \emptyset \end{cases},$$

where $\{NE_s\}$ is the set of NEs in the current sentence, and $\{NE_{title}\}$ is the set of NEs in the title. For example, if "IL-2" both exists in the title and current sentence, the value of f_{bNE} will be 1, otherwise 0. The second, f_{NE}, is the number of appearance of shared NEs in the current sentence.

WF Feature

This feature is designed for evaluating the importance of a word in the result section. Under our approach, a word's importance is defined as the ratio of its frequency in the result section over its frequency in the other sections. Therefore, we use this ratio to select the most important words. The ratios of all words are calculated based on the training set. For example, "significantly" appears 3,514 times in the result section but only 36 times in the other sections. Thus, its ratio will be 98.99% (3,514 over 3,550). All words with ratios higher than 80% are put in the candidate list, which our in-lab biomedical researchers examine manually to remove unimportant words.

On our candidate list, the top ten important words are "conclude", "significantly", "showed", "resulted", "pronounced", "statistically", "significant", "higher", "decreased", and "similar", as shown in Table 1.

Table 1. Important words.

Word	Only in Result (times)	In abstract (times)	Ratio (%)
conclude	99	100	99
significantly	3514	3550	98.99
showed	1036	1064	97.37
resulted	173	178	97.19
pronounced	132	136	97.06
statistically	381	400	95.25
significant	2607	2746	94.94
higher	2542	2678	94.92
decreased	1265	1337	94.61
similar	1065	1165	91.42

4. Results

4.1. Datasets

Several biomedical researchers select approximately six thousand PubMed abstracts related to hypertension and gene as our biomedical corpus. To evaluate the performance of our system, we randomly selected two thirds of the abstracts as the training set and used remaining as the test set. Table 2 shows the two datasets.

Table 2. Hypertension-gene relation corpus.

Dataset	Quantity
Training dataset	3808
Test dataset	1903

After preparing the corpus, our biologists manually labeled the result section for each abstract in the corpus.

4.2. Experiment Measurement

We use three measurements, precision, recall, and F-Measure, to evaluate the performance of our result identification system. The metrics are defined as follows:

$$\text{Precision} = \frac{\text{Predicting Result} \cap \text{True Answers}}{\text{Predicting Result}}$$

$$\text{Recall} = \frac{\text{Predicting Result} \cap \text{True Answers}}{\text{True Answers}}$$

$$\text{F-Measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

4.3. Experiment Design and Result

We design seven configurations to assess the effectiveness of each feature type and compare the performance of the following feature combinations: NE + WF, Position + NE, Position + WF, and Position + NE + WF. Table 3 shows the evaluation results.

Table 3. The evaluation results. P stands for Precision, R stands for Recall and F stands for F-Measure.

Feature Set	P (%)	R (%)	F (%)
Position	87.08	93.06	89.97
NE	97.85	60.42	74.71
WF	94.23	65.30	77.14
NE + WF	94.36	65.21	77.12
Position + NE	93.16	87.01	89.98
Position + WF	94.89	88.68	91.68
Position + NE + WF	95.32	89.85	92.50

As shown in Table 3, configurations with single NE and WF features yielded comparatively low recall and F-Measure scores. This demonstrates that the Position feature is the most effective feature both for precision and recall. We observe that the NE and WF features only achieve comparatively high precision rates. Moreover, combining the Position feature with NE or WF improves the precision rate substantially (6.08% for Position + NE and 7.81% for Position + WF), without hurting the F-Measure too much.

In our experiment, the best result was obtained by the combination of the three proposed feature sets, Position + NE + WF. It achieved F-measure 92.50%, Precision 95.32% and Recall 89.85%.

The best result of our experiment is the combination of the proposed three feature sets, Position + NE + WF. It achieves F-measure 92.50%, Precision 95.32% and Recall 89.85%. The result shows our proposed feature sets could effectively identify sentences belonging to result section.

5. Discussion

In this section, we discuss the effects of three feature sets to justify our experimental results and compare Position only with Position +NE. Finally, we explain why we choose CRFs as our ML model in more detail.

5.1. The effects of Position, NE, and WF

5.1.1. Position

Since the structure of biomedical abstracts is fairly uniform, the position feature is the most important feature for result identification. However, in some special cases, the position feature alone cannot correctly disambiguate the result section. For example, two abstracts may contain different numbers of sentences in total, and their result sections may also contain different numbers of sentences. These differences affect the position feature's prediction capability slightly; hence, we need to incorporate more features, such as NE and WF, into our CRF model.

5.1.2. Named Entity and Word Frequency

These two feature sets have the same characteristic in that when sentences in an abstract have frequent words (or NEs that also appear in the title), WF (or NE) feature will be enabled; otherwise it will be disabled. We design these two features, which should only be enabled in the sentences of the result section. Therefore, if the title's NEs do not appear in the result section, the NE feature would be useless. For WF, it has the same condition.

Our experiments show that when NE (or WF) feature is enabled, the proposed method yields a high precision rate of 94.23% for WF and 97.85% for NE. However, they both suffer the low recall because in our corpus (contains abstracts), not all sentences from result sections of abstracts have NEs or frequent words. This explains why low recall is.

5.2. Position + NE v.s. Position only

In our test set, the abstract (PMID: 18269635) has a sentence "*..., we have focused on the potential vasculoprotective effects of both **IGF-I** and IGFBP-1.*" in the result section with a NE, IGF-I. The title of the abstract "*The role of **IGF-I** and its binding proteins in the development of type 2 diabetes and cardiovascular disease.*" also has the same NE. The evaluation result showed that only applying Position feature will lead to identification error. We explain the reason via the following example.

Two abstracts have the same total number of sentences, but one has six sentences in result section and the other has four sentences. This case may lead CRFs to identify incorrectly because the CRFs only depend on the distribution of the Position feature and the boundary tag in training data, as mentioned in Section 5.1.1. However, after introducing the NE feature, our system has more information to determine whether the sentence is in the result section. In our example, the sentence contains an NE that also occurs in the title; thus, the CRFs can successfully identify the sentence in the result section.

5.3. CRFs v.s. Other ML Models

In this section, we explain why we choose the CRFs as our ML model.

One way to solve the section identification problem is to use the classifier-based approaches, such as Support Vector Machines [14] or Maximum Entropy [15]. In these approaches, we can take each sentence as a tag class. For result section identification, we can use the training data to train a binary classifier and apply it to determine whether each sentence belongs to result section. However, there are some problems in this approach. The most obvious flaw is that using a binary classifier to process all the sentences in an abstract causes the identified result to become segmented into many pieces. Therefore, we need a classifier that can assign a class to each sentence in sequence.

The left-to-right classifier may resolve this problem. When classifying each sentence we can rely on features from the current sentence, and the output of the classifier from previous sentence. While this technique seems to solve the problem, it makes a hard decision about each sentence before moving on to the next sentence. Hence, the classifier is unable to use information from subsequent sentences.

The Maximum Entropy Markov Model (MEMM) [16] is an augmentation of the basic ME model that incorporates the Viterbi algorithm into ME. MEMM addresses the problem of Hidden Markov Models (HMM) [17] in that HMM lies in data sparseness problem and it inappropriately uses a generative joint model to solve a conditional problem as described by Tsai et al. [5]. However, MEMM still has a label bias problem: the Markov assumptions make the transitions of MEMM leaving a given state compete only against each other, rather than against all other transitions in the model [5, 11]. Therefore, we use the CRFs model introduced by Lafferty et al. [11] to avoid the label bias problem and propose the formulation describing in Section 3.1 to transform the section identification problem into a sequential tagging problem which can be solved by CRFs.

6. Conclusion

The result and conclusion sections are an important part for biomedical research. In this paper, we utilized conditional random fields (CRFs) and three proposed feature sets to solve the result identification problem. Our experiments showed three important results. Firstly, we

showed the position of the sentence is the most important feature for result identification. Secondly, we demonstrate that the named entities (NEs) information of the title can be incorporated into the ML model to further improve the precision. Finally, the selected words which frequently appeared in the result section can help the ML model identify the result section better.

In the future work we plan to (1) Add more features to improve the current model. (2) Apply this methodology to different field, not just biomedical field.

7. Acknowledgement

This research was supported in part by the National Science Council under Center of Excellence Grant NSC 96-2752-E-001-001-PAE and thematic program of Academia Sinica under Grant AS95ASIA02.

8. References

[1] K. Fukuda, A. Tamura, T. Tsunoda, and T. Takagi, "Toward information extraction: identifying protein names from biological papers," *Pacific Symposium on Biocomputing,* pp. 707-718, 1998.

[2] S. Pyysalo, F. Ginter, J. Heimonen, J. Bjorne, J. Boberg, J. Jarvinen, and T. Salakoski, "BioInfer: a corpus for information extraction in the biomedical domain," *BMC Bioinformatics,* vol. 8, p. 50, 2007.

[3] K. Jin-Dong, O. Tomoko, Y. T. Yoshimasa Tsuruoka, and N. Collier, "Introduction to the bio-entity recognition task at JNLPBA," *Proceedings of the International Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA-04),* pp. 70–75, 2004.

[4] B. Settles, "Biomedical Named Entity Recognition Using Conditional Random Fields and Rich Feature Sets," in *COLING 2004 International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA)*, 2004.

[5] R. T.-H. Tsai, C.-L. Sung, H.-J. Dai, H.-C. Hung, T.-Y. Sung, and W.-L. Hsu, "NERBio: using selected word conjunctions, term normalization, and global patterns to improve biomedical named entity recognition," *BMC Bioinformatics,* vol. 7 Suppl 5, p. S11, 2006.

[6] P. Ruch, C. Boyer, C. Chichester, I. Tbahriti, A. Geissbuhler, P. Fabry, J. Gobeill, V. Pillet, D. Rebholz-Schuhmann, and C. Lovis, "Using argumentation to extract key sentences from biomedical abstracts," *International Journal of Medical Informatics,* vol. 76, pp. 195-200, 2007.

[7] J. Lin, D. Karakos, D. Demner-Fushman, and S. Khudanpur, "Generative Content Models for Structural Analysis of Medical Abstracts," *Proceedings of the BioNLP Workshop on Linking Natural Language Processing and Biology at HLT-NAACL,* vol. 6, pp. 65-72, 2006.

[8] J. C. Wu, Y. C. Chang, H. C. Liou, and J. S. Chang, "Computational analysis of move structures in academic abstracts," *Proceedings of the COLING/ACL on Interactive presentation sessions,* pp. 41-44, 2006.

[9] Y. Yamamoto and T. Takagi, "A Sentence Classification System for Multi Biomedical Literature Summarization," *Proceedings of the 21st International Conference on Data Engineering,* 2005.

[10] M. Shimbo, T. Yamasaki, and Y. Matsumoto, "Using sectioning information for text retrieval: a case study with the MEDLINE abstracts," *In Proceedings of Second International Workshop on Active Mining (AM'03),* 2003.

[11] J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," in *ICML-01*, 2001, pp. 282-289.

[12] L. Rabiner and I. A. W. a. K.-F. Lee, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," in *Readings in Speech Recognition*, A. Weibel and K.-F. Lee, Eds., 1990, pp. 267–296.

[13] F. Sha and F. Pereira, "Shallow parsing with conditional random fields," *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1,* pp. 134-141, 2003.

[14] J. Kazama, T. Makino, Y. Ohta, and J. Tsujii, "Tuning support vector machines for biomedical named entity recognition," in *ACL-02 Workshop on Natural Language Processing in Biomedical Applications*, 2002.

[15] H. L. Chieu and H. T. Ng, "A maximum entropy approach to information extraction from semi-structured and free text," *Proceedings of the Eighteenth National Conference on Artificial Intelligence,* pp. 786–791, 2002.

[16] A. McCallum, D. Freitag, and F. Pereira., "Maximum entropy Markov models for information extraction and segmentation," in *ICML' 00*, 2000.

[17] S. Zhao, "Named Entity Recognition in Biomedical Texts using an HMM Model," in *COLING 2004 International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA)*, 2004.