A Multi-level Disambiguation Framework for Gene Name Normalization

SUN Cheng-Jie[1] WANG Xiao-Long[1] LIN Lei[1]
LIU Yuan-Chao[1]

Abstract The flexible nomenclature of gene name results in severe semantic ambiguity, which is an obstacle for deep biomedical text mining. Gene name normalization (GN) is an effective way to resolve this problem. In this work, a multi-level disambiguation framework was proposed to solve gene name normalization problem. Aiming at different ambiguity situations during the procedure of GN, three different strategies were included in the framework. They were dictionary-based gene name detection, machine-learning-based candidate selection, and semantic-based disambiguation. Experimental results showed that the proposed method could achieve 0.746 *F*-measure on the BioCreAtIvE2006 GN task test data set.

Key words Gene name normalization (GN), maximum entropy model, semantic similarity

With the rapid development in biological and biomedical research field, a large number of research papers and on-line databases have been produced. For example, the primary bibliographic database MEDLINE contains 16 million references to journal articles and over 2 000 new references are added each day. There is an urgent need for efficient and effective solutions for making effective use of the knowledge resource, which makes biomedical text mining a hot research field[1−2].

The purpose of gene name normalization (GN) was to correctly associate the gene names in documents with standard identifiers. GN has the following benefits: 1) improving the work efficiency of database curators[3]; 2) helping biology researchers to get more accurate information and to analyze and summarize bodies of the published works[4]; 3) providing necessary informative knowledge for further text mining task, such as relation extraction[5].

BioCreAtIvE (Critical assessment for information extraction in biology) has held two competitions to highlight GN task. These two competitions resulted in many novel and useful approaches, but the results clearly identified that more important work is necessary[6−7]. The flexible nomenclature is the primary reason for the challenges of GN. The flexible nomenclature is mainly caused by the long history of biomedical research and the lack of inter-species naming conventions. Even though guidelines are available for human gene, such as nomenclature from the human genome organization (HUGO), [8] showed that the scientific community had not widely adopted the guidelines, and there was no clear tendency that this situation was improving. Currently, one gene could be referred to as several different names (synonymy), and a name could be associated with multiple gene identifiers (ambiguity). For example, in the dictionary afforded by BioCreAtIvE2006, each unique gene identifier has 5.55 synonyms on average, while each synonym has 1.12 gene identifiers on average.

The approaches to GN could be roughly classified into three classes: pattern-matching-based approaches[5, 9], machine-learning-based methods[10], and combinations of these approaches[3, 11]. Usually an integrated system for GN includes three steps: 1) detecting mentioned gene name, 2) identifying the semantic intent of each mention, and 3) associating each mention with the right gene identifier. Some researches only focus on one of these steps, for example, Tsuruoka[12] utilized logistic regression method to improve the accuracy of gene name detection; Xu[13] proposed a knowledge-based method to build a system for Step 3), i.e., gene name disambiguation.

In this work, a multi-level disambiguation framework was proposed to solve GN problem. We used dictionary match strategy to detect gene names and then associate them with standard identifiers. Dictionary match strategy could produce ambiguities because a match: 1) may be a general English word, due to the existence of lots of common English words in the dictionary, such as "of", "can", and "end"; 2) may be a gene referred to as other species; and 3) may denote a RNA or a protein. Maximum entropy (ME) model was adapted to judge whether a match is meaningful using the contexts around the match and the orthography features of the match. Then, the meaningful matches were associated with the standard identifiers assigned in the dictionary. There are still lots of ambiguities in the meaningful matches as a name could be associated with multiple gene identifiers. Knowledge-based disambiguation strategies were proposed to solve them. The knowledge-based strategies include human writing habit filter, inverse document frequency (IDF) filter, and semantic similarity based disambiguation. We achieved an *F* score of 0.746 on the BioCreAtIvE2006 GN evaluation data set.

The rest of this paper is organized as follows. Section 1 describes the GN task in detail. In Section 2, we propose our multi-level disambiguation framework for GN. Section 3 presents the experiment results and analysis. Section 4 concludes this work.

1 Problem definition

The goal of GN is to correctly associate the gene names in documents with standard identifiers. For example, the following sentence "Molecular cloning of the cDNA for human TrkC (NTRK 3), chromosomal assignment, and evidence for a splice variant" contains two gene names "TrkC" and "NTRK 3". Both of the two gene names are associated with ID "4 916" in Entrez Gene database.

So firstly, a gene name detection procedure is required for GN task. In above example, gene name "TrkC" and "NTRK3" should be correctly recognized. But this procedure is different from a new term detection because GN task needs to utilize not only new gene names but also existing gene names. A new term detection system tries to find new gene names, which have not been embodied in the corresponding databases; while a gene name normalization system always needs the gene identifiers from gene name databases. Thus, dictionary-based approaches are more appropriate for GN task than that for new term detection task.

Besides, GN task has to choose a right ID (sense) for a given gene name according to its context. A gene name may correspond to multiple IDs. For example, the gene name "NTRK3" has ten IDs in the Entrez Gene database, including "18 213", "4 916", "29 613", and so on. But according to its context, its correct ID is "4 916" in the above

Received December 12, 2007; in revised form June 13, 2008
Supported by National High Technology Research and Development Program of China (863 Program) (2006AA01Z197, 2007AA01Z172) and National Natural Science Foundation of China (60673019, 60435020)

1. School of Computer Science, Harbin Institute of Technology, Harbin 150001, P. R. China

DOI: 10.3724/SP.J.1004.2009.00193

example. So compared with the new term detection, GN is syntactically easier because identification of the textual boundaries of each mention is not required. However, GN poses significant semantic challenges, as it requires detection of the actually intended gene, along with reporting the gene in a standardized form[10]. Semantic disambiguation is necessary for a GN system because ambiguity phenomena are very serious for gene names as shown in Table 1. Even if it was possible to identify every gene mention with 100 % accuracy, it would still be difficult to disambiguate each mention given the number of possibilities and the high degree of overlap among synonym lists for different but related genes.

Table 1 Statistics of gene names in different dictionaries provided by BioCreAtIvE

Dictionary	Number of IDs	Average number of synonyms per ID	Average number of IDs per synonym
Human	32 975	5.5	1.12
Fly	27 749	2.944	1.09
Yeast	7 928	1.861	1.01
Mouse	52 594	2.482	1.02

2 Method

In this work, a multi-level disambiguation framework was proposed for GN task. The proposed framework includes 3 main components: matching, candidate selection, and ambiguity resolving. In the following three subsections, we will describe the three components, respectively. The whole flowchart of our framework is shown in Fig. 1.

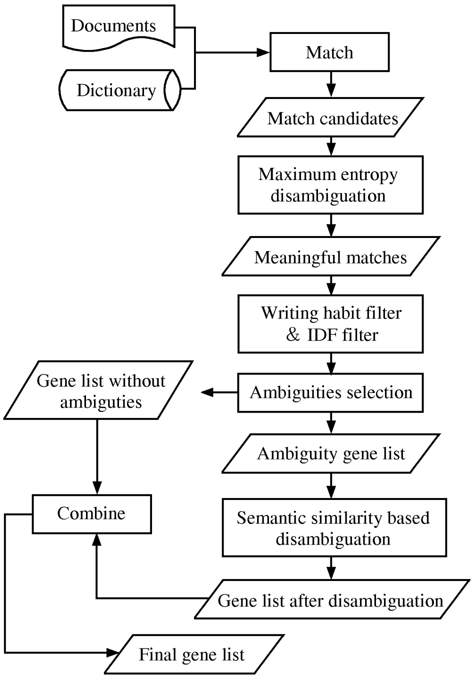
Documents
Dictionary
Match
Match candidates
Maximum entropy disambiguation
Meaningful matches
Writing habit filter & IDF filter
Gene list without ambiguties
Ambiguities selection
Ambiguity gene list
Combine
Semantic similarity based disambiguation
Gene list after disambiguation
Final gene list

Fig. 1 The flowchart of the proposed framework

2.1 Dictionary-based matching

As mentioned previously, the target of GN task is to correctly associate the gene names in documents with standard identifiers. No large effort for finding new gene name is demanded, so a dictionary-based matching strategy could better suit to GN task. Usually, the standard identifiers are assigned in some databases, such as Entrez Gene, FlyBase, and HUGO. Dictionaries including the standard identifier and its corresponding synonym could be constructed using the existing databases. In our framework, matching is a procedure for matching words in text against terms in the dictionary. Its purpose is to find the gene name candidates. This required preprocessing both the words in the text and the words or terms in the dictionary. We used a dictionary built from the lexical resource provided by BioCreAtIvE2006 evaluation GN task. Stop words were removed from the dictionary. A regular expression pattern is compiled for each item in the dictionary to handle the case variation, which could improve the recall of the system. Simple tokenization was put on the text. A blank was added prior to punctuations in order to increase the number of matches. Also abbreviated enumerations of multiple gene names were replaced with their respective fully expanded forms. For example: 1) IL3/IL5 → IL3 and IL5; 2) freac1-freac7 → freac1, freac2, $\cdots$, freac7. Stemming was not applied because experimental results showed that it could decrease the system performance as mentioned in [5].

2.2 Candidate selection by ME model

Inspired by [10], we want to build a model that, given a set of synonym matches, distinguishes meaningful ones from unmeaningful ones. Here, meaningful matches refer to the matches which are the true name candidates, not false gene names, such as some common English words. This is essentially a binary classifier. Maximum entropy (ME) model was chosen to do the classification task in our system. ME model is defined as

$$p(y|\boldsymbol{X}) = \frac{1}{Z(\boldsymbol{X})}\prod_{i=1}^{K}\lambda_i^{f_i(\boldsymbol{X},y)}$$
$$Z(\boldsymbol{X}) = \sum_y p(y|\boldsymbol{X}) = \sum_y\prod_{i=1}^{K}\lambda_i^{f_i(\boldsymbol{X},y)} \tag{1}$$

where y is a class label (in our case: meaningful or unmeaningful), $\boldsymbol{X}$ is an input vector containing predicates on the matched text, and $Z(\boldsymbol{X})$ is a normalization term. Each feature function $f_i(\boldsymbol{X}, y)$ maps an input vector and class to a binary value, for example,

$$f_i(\boldsymbol{X}, y)=\begin{cases}1, & \text{if currentword} = SYT \text{ \& } y = \text{meaningful}\\ 0, & \text{otherwise}\end{cases} \tag{2}$$

The parameters of the model are the feature weights λ_i. They are determined in such a way that the parameters maximize the conditional log-likelihood of the training data $\sum_{k=1}^{N}\log p_{\boldsymbol{\lambda}}(y^{(k)}|\boldsymbol{X}^{(k)})$, where N is the number of training samples. We used Zhang′s ME tool[14] to train the ME model.

Our work are different from [10] in two ways: 1) More features are involved including context features and the orthography features in our ME model; 2) We just used ME model to select the meaningful matches (the answer candidates), not the answers as in [10]. Further knowledge-based disambiguation is done to the meaningful matches.

The training data for the classifier were collected from the dictionary-based matching results of the BioCreAtIvE2006 GN task training set. If the normal form for a match was in the normalized gene list for that document, then the match was labeled meaningful; otherwise, it was labeled unmeaningful. This provided a large set of meaningful and unmeaningful matches required to train an ME classifier. For each match, the text that matched, the three words right before the match, the three words right after the match ([10] used two words both before and after the match), and the normal form causing the match were extracted as features. Besides, several orthography features were extracted for the match, such as whether the letters in the matched term are all upper-case, whether the matched term contains digits. These features could help to detect whether a match is a gene name as in gene named entity recognition task[15]. When testing, the system first extracted all the matches that occurred within test set. Then, for each match, the classifier would judge whether it is meaningful or unmeaningful according to its features.

2.3 Knowledge-based disambiguation

Through previous steps, we could get lots of gene name candidates. Some false gene names still exist because ME model could not filter the false gene names in one hundred percent. Besides, semantic ambiguities will appear when assigning standard identifiers (ID) to gene names because one gene name may correspond to several identifiers in the dictionary. Here, three knowledge-based disambiguation methods, including inverse document frequency (IDF) filter, writing habit filter, and semantic similarity-based disambiguation, are used to further remove the false gene names and to resolve the semantic ambiguities.

1) IDF value filter

IDF could estimate the importance of a term in a given document[16]. The IDF value of a term t is defined as

$$\mathrm{IDF}(t)=\log\left(\frac{\text{Number of all documents}}{\text{Number of documents which contain term } t}\right) \quad (3)$$

For a term t, a small IDF value means that it occurs in a lot of documents, so it is unlikely to be a meaningful or specific gene name. Thus, we could use IDF value to filter the false candidates. The IDF value of each term was calculated according to the noisy training data of BioCreAtIvE2006 GN task.

2) Writing habit filter

There are two principles for word sense disambiguation: a) one sense per collocation (i.e., assign a single ID for each name within a document); b) one sense per discourse (i.e., assign the same ID to all instances of a given name within a document)[17]. In this work, we adopted these two principles and proposed a filter method to get rid of some false positive answers. The basic idea is that the more names an ID has in a document, the more probability it is a true positive ID. So each time, we chose the ID with the most corresponding names (at least 2) as the correct ID, and removed its corresponding names from the name list of other IDs.

3) Semantic similarity based disambiguation

From the gene list gotten by previous steps, we could get a list consisting of ambiguity gene names by choosing the names, which have more than one identifier. For example, SYT can refer to two human genes with different identifiers, SYT1 (ID 6 857) and SS18 (ID 6 760).

We proposed a disambiguation method based on semantic similarity. A profile was built for each ID in the ambiguity name list. For each ambiguity name, the similarity between the profiles of its corresponding IDs and current document that contains the name was calculated. The ID with the highest similarity score was chosen as the ID of the name. The content of the profile for each ID actually includes the PubMed IDs (PID) of the documents related to the gene ID from gene2PubMed file[1]. The gene2PubMed file records the PIDs of the documents related to each gene ID, which accumulates the expert knowledge for gene annotation. For example, Gene ID 63 976 (MEL1) has 14 related documents and their PubMed IDs are listed in Table 2. The way we utilized this knowledge source was to find the semantic similarity between the new document and the annotated documents. For a name with multiple IDs in the new document, we will the choose ID with highest similarity score as its correct ID.

Table 2 The profile for ID 63 976

ID	PID
63 976	4 063 527, 8 547 101, 11 050 005, 11 214 970, 12 168 954, 12 477 932, 12 557 231, 12 816 872, 14 656 887, 14 702 039, 14 712 237, 16 582 916, 16 598 304, 16 637 659

Thus, we come to the problem how to calculate the semantic similarity between two documents. WordNet[18], a lexical database which is available online and provides a large repository of English lexical items, was used as the resource for semantic similarity calculation. Based on WordNet, a semantic approach was proposed[19], which has the following steps (Here, we consider a document as a long sentence): 1) Tokenization; 2) Find the most appropriate sense for every word in a sentence; 3) Compute the similarity of the sentences based on the similarity of the pairs of words.

For Step 2), a modified Lesk algorithm[20] was adapted[21]. The new algorithm involves more contexts and knowledge than the original Lesk and applies a new scoring mechanism to measure gloss overlap that gives a more accurate score than the original Lesk bag of words counter. For Step 3), the similarity of the pairs of words is calculated according to their position in WordNet. If any of them does not occur in WordNet, the edit distance is calculated as their similarity.

The task of calculating two sentences similarity could be formulated as the problem of computing a maximum total matching weight of a bipartite graph, where the tokens in the two sentences could be considered as two sets of disjointed nodes, denoted by X and Y. In practice, a greedy algorithm is used considering the time efficiency. The final similarity score is gotten through matching average, which is defined as

$$Sim(S1, S2) = \frac{2 \times Match(X,Y)}{|X| + |Y|} \quad (4)$$

where $Match(X,Y)$ are the matching word tokens between X and Y. This similarity is computed by dividing the sum of similarity values of all match candidates of both sentences by the total number of tokens. An important point is that it is based on each of the individual similarity values, so that the overall similarity always reflects the influence of them.

3 Experiment

The purpose of our experiments lies in two aspects: 1) to show the effect of each step in the proposed framework; 2) to indicate the semantic similarity-based disambiguation is very effective.

[1] ftp://ftp.ncbi.nih.gov/gene/DATA/gene2PubMed.gz

3.1 Data set

Due to the difficulties of obtaining large quantities of full text articles, only the abstracts of articles from MEDLINE are available for the evaluation of gene name normalization in GN research field currently.

The data set of BioCreAtIvE2006 gene normalization task was chosen to evaluate our method. This data set used the GOA annotated records as the basis for selecting documents rich in human genes and proteins. However, the GOA annotators annotate full text, and we were using only abstracts; furthermore, the GOA annotation process does not include every human gene mentioned in an article, but only specific genes of interest. So, further annotations were done by experts. This data set includes three parts: training data, test data, and noisy data. Their detail information is shown in Table 3.

Table 3 The statistics information of the experimental data

Data set	Number of abstracts	Annotation
Training data	281	Annotated by Expert
Test data	262	Annotated by Expert
Noisy data	5 000	Annotated automatically

3.2 Evaluation

The performance of GN was measured using precision, recall, and F-measure. The results were computed based on a simple matching of gene identifiers against the gold standard for each abstract. Identifiers that matched the gold answers constituted true positives (TP), identifiers that did not match were false positives (FP), and gold standard identifiers that were not matched were false negatives (FN). Recall(R), precision(P), and F-measure (F) were computed in the usual way: $R = TP/(TP + FN)$; $P = TP/(TP + FP)$; $F = 2 \times P \times R/(P + R)$.

3.3 Experimental results and analysis

The evaluations were done on BioCreAtIvE2006 test data set. The whole results were shown in Table 4.

Table 4 Experimental results on Biocreative2006 GN test data

Step	P	R	F
Dictionary-based matching	0.311	0.859	0.457
+ME-based disambiguation	0.506	0.831	0.629
+writing habit filter	0.525	0.829	0.643
+IDF filter	0.563	0.805	0.663
+remove ambiguous gene names	0.760	0.703	0.730
+semantic based disambiguation	0.717	0.778	0.746

The first row in Table 4 gives the performance of dictionary-based matching component. As expected, simple dictionary-based matching achieved high recall but also produced lots of false positive instances. Two kinds of reasons caused false positive instances: 1) ambiguity between gene names and common English words and domain-related terms; 2) annotation process does not include every gene mentioned in an abstract but only specific genes of interest. Many systems participated in BioCreAtIvE2006 GN task also used dictionary-based matching method to get the gene name candidates. The best system, which achieved an F-measure of 0.81, used a more comprehensive dictionary and more complex regular expression.

ME-based disambiguation method was used to filter the false positive instances (unmeaningful). Context features and orthography features were considered in our ME model as described in Section 2.2. The results were shown in the second row of Table 4, where we can see that ME-based disambiguation can dramatically increase the precision to 0.506 from 0.311 by causing less impact on the recall.

The knowledge-based disambiguation includes 4 steps: writing habit filter, IDF filter, ambiguity filter, and semantic similarity disambiguation. Although the writing habit filter did not reduce the number of false positive severely, it kept the number of true positive almost unaltered. So, it is still very useful for the GN task. The IDF filter utilized a gene name list to filter the false positive names. The gene name list consists of gene name selected from the answer candidates according to the following conditions: 1) IDF value less than 4.0; 2) the case where there are no numbers. The IDF value of each gene name is calculated with noisy training data. With IDF filter, the F-measure increased by 2 %.

A gene name is ambiguous if it associates a mention with multiple gene identifiers. In this case, the ambiguity filter removes all the ambiguous gene names from the current answer list. With this filter, an ambiguous gene name list could be generated. In our experiments, we got a list consisting of 143 gene names, which contained 77 possible true positives. Removing ambiguous gene names could generate a high precision result because it reduced the number of false positive instances a lot.

Semantic similarity-based disambiguation strategy was used to select the right gene identifier for each name in the ambiguous list. About 13 000 abstracts used for semantic similarity calculating were downloaded using the tool provided by Biopython. Note that PIDs of the abstracts which contain the ambiguous names have been excluded from the corresponding profiles. The accuracy of disambiguation in this step is 76.6 % (59 out of 77 possible true positives were correctly recognized). Xu[13] utilized the vector space model to calculate the similarity between the context vector of a name and the profile vector of an ID, and achieved an accuracy of 77.8 % for fly gene name disambiguation. It is difficult to compare the two results since the data sets are different. But from Table 1, we can see human gene name is more ambiguous than fly gene name. In this way, our disambiguation result for human gene name is a promising result.

From Table 4, we can see that the proposed framework can make use of various techniques to resolve the different ambiguous situations during the whole GN process and improve the performance step by step.

3.4 Comparison with related works

In order to show the effect of the proposed framework, we also compared our results with the performances of the 20 participants in BioCreAtIvE2006. Our system can achieve rank 10 according to the F-measure. The best F-measure performance is 0.810 achieved by Team 042 (T042)[22] as shown in the first row of Table 5. T042 also used dictionary-based matching method to find the gene name candidates, but with an extended synonym list, which would help to get high recall. We could not repeat the synonym list extending process because no detail information was given in [22]. If with the same synonym list as ours, the F-measure of T042 is 0.716 as shown in the second row of Table 5, which is lower than our result 0.746. Besides, T042 used manually building rules to filter the FP instances while we used a machine learning method to do this automatically. The third and the fourth rows in Table 5 show that the two

Table 5 Performance comparison with the best system in BioCreAtIvE2006 GN task

System	Run setting	P	R	F
T042	With extended synonym list, $FP + FN$ filter, disambiguation	0.789	0.833	0.810
T042	With un-extended synonym list, FP filter, disambiguation	0.707	0.725	0.716
T042	With extended synonym list and manually building rules	0.496	0.875	0.633
Our system	With un-extended synonym list and ME method	0.506	0.831	0.629

methods got comparable results. From Table 5, one can conclude that the performance promotion of T042 was mainly given by the extend synonym list and the FN filter. Although T042 achieved better results than our GN system in the whole, our system is easier to be ported to other organisms because it needs less resources and it is more automatic.

4 Conclusion

In this work, we proposed a multi-level disambiguation framework for gene name normalization task. An F-measure of 0.746 was achieved in the BioCreAtIvE2006 GN task test data set with the proposed framework. We have three contributions:

1) Design a way to combine multi-source knowledge to complete GN task;

2) Show the effect of ME model to filter the false positive instances with context features and orthography features;

3) Propose a semantic similarity-based disambiguation method.

This method can use the cumulate annotation knowledge in the annotated documents, thus it could be very useful in practical gene annotation.

References

1 Jensen L J, Saric J, Bork P. Literature mining for the biologist: from information retrieval to biological discovery. *Nature Reviews Genetics*, 2006, **7**(2): 119−129

2 Nenadic G, Spasic I, Ananiadou S. Terminology-driven mining of biomedical literature. *Bioinformatics*, 2003, **19**(8): 938−943

3 Morgan A A, Hirschman L, Colosimo M, Yeh A S, Colombe J B. Gene name identification and normalization using a model organism database. *Journal of Biomedical Informatics*, 2004, **37**(6): 396−410

4 Yetisgen-Yildiz M, Pratt W. Using statistical and knowledge-based approaches for literature-based discovery. *Journal of Biomedical Informatics*, 2006, **39**(6): 600−611

5 Fang H, Murphy K, Jin Y, Kim J S, White P S. Human gene name normalization using text matching with automatically extracted synonym dictionaries. In: Proceedings of the BioNLP Workshop on Linking Natural Language Processing and Biology at HLT-NAACL. New York, USA: New York University, 2006. 41−48

6 Hirschman L, Colosimo M, Morgan A A, Yeh A. Overview of biocreative task 1b: normalized gene lists. *BMC Bioinformatics*, 2005, **6**(1): 11

7 Morgan A A, Hirschman L. Overview of biocreative II gene normalization. In: Proceedings of the 2nd BioCreative Challenge Evaluation Workshop. Madrid, Spain: CNIO, 2007. 17−28

8 Tamames J, Valencia A. The success (or not) of hugo nomenclature. *Genome Biology*, 2006, **7**(5): 402

9 Hanisch D, Fundel K, Mevissen H, Zimmer R, Fluck J. Prominer: rule-based protein and gene entity recognition. *BMC Bioinformatics*, 2005, **6**(1): 14

10 Crim J, McDonald R, Pereira F. Automatically annotating documents with normalized gene lists. *BMC Bioinformatics* 2005, **6**(1): 13

11 Fundel K, Güttler D, Zimmer R, Apostolakis J. A simple approach for protein name identification: Prospects and limits. *BMC Bioinformatics*, 2005, **6**(1): 15

12 Tsuruoka Y, McNaught J, Tsujii J, Ananiadou S. Learning string similarity measures for gene/protein name dictionary look-up using logistic regression. *Bioinformatics*, 2007, **23**(20): 2768−2774

13 Xu H, Fan J W, Hripcsak G, Mendonca E A, Markatou M, Friedman C. Gene symbol disambiguation using knowledge-based profiles. *Bioinformatics*, 2007, **23**(8): 1015−1022

14 Zhang L. Maximum entropy modeling [Online], available: http://homepages.inf.ed.ac.uk/s0450736/maxent.html, May 10, 2008

15 Tsai T H, Chou W C, Wu S H, Sung T Y, Hsiang J, Hsu W L. Integrating linguistic knowledge into a conditional random field framework to identify biomedical named entities. *Expert Systems with Applications*, 2006, **30**(1): 117−128

16 Salton G, Buckley C. Term-weighting approaches in automatic text retrieval. *Information Processing and Management: an International Journal*, 1988, **24**(5): 513−523

17 Yarowsky D. Unsupervised word sense disambiguation rivaling supervised methods. In: Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics. Cambridge, USA: Association for Computational Linguistics, 1995. 189−196

18 Fellbaum C. *Wordnet: an Electronic Lexical Database*. Cambridge: The MIT Press, 1998. 1−9

19 Simpson T, Dao T. WordNet-based semantic similarity measurement [Online], available: http://www.codeproject.com/cs/library/semanticsimilaritywordnet.asp, December 20, 2007

20 Lesk M. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In: Proceedings of the 5th Annual International Conference on Systems Documentation. New York, USA: ACM, 1986. 24−26

21 Banerjee S, Pedersen T. Extended gloss overlaps as a measure of semantic relatedness. In: Proceedings of the 18th International Joint Conference on Artificial Intelligence. Acapulco, Mexico: KFUPM Eprints, 2003. 805−810

22 Hakenberg J, Royer L, Plake C. Me and my friends: gene mention normalization with background knowledge. In: Proceedings of the 2nd BioCreative Challenge Evaluation Workshop. Madrid, Spain: CNIO, 2007. 141−144

SUN Cheng-Jie Ph. D. candidate at the School of Computer Science, Harbin Institute of Technology. His research interest covers text mining and information extraction. Corresponding author of this paper. E-mail: cjsun@insun.hit.edu.cn

WANG Xiao-Long Professor at Harbin Institute of Technology. His research interest covers artificial intelligent, natural language processing, information retrieval, and bioinformatics. E-mail: wangxl@insun.hit.edu.cn

LIN Lei Associate professor at Harbin Institute of Technology. His research interest covers artificial intelligent, text mining, and bioinformatics. E-mail: linl@insun.hit.edu.cn

LIU Yuan-Chao Associate professor at Harbin Institute of Technology. His research interest covers text clustering, text summarization, and artificial neural networks.
E-mail: lyc@insun.hit.edu.cn