行政院國家科學委員會專題研究計畫　成果報告

資訊萃取技術在生物醫學文獻上的應用與探討(2/2)

計畫類別：個別型計畫
計畫編號：NSC94-2213-E-009-024-
執行期間：94 年 08 月 01 日至 95 年 07 月 31 日
執行單位：國立交通大學資訊科學學系(所)

計畫主持人：梁婷

計畫參與人員：吳典松 朱俊榮 施並格 林裕祥 黃立泓　施曉茹 蘇傳堯

報告類型：完整報告
報告附件：出席國際會議研究心得報告及發表論文
處理方式：本計畫可公開查詢

中　華　民　國 95 年 9 月 18 日

行政院國家科學委員會補助專題研究計畫 **ˇ 成 果 報 告**
□期中進度報告

(計畫名稱)

資訊萃取技術在生物醫學文獻上的應用與探討

計畫類別：ˇ 個別型計畫　　□ 整合型計畫
計畫編號：NSC 94 － 2213 －E －009 －024 －
執行期間： 94 年 08 月 01 日至 95 年 07 月 31 日

計畫主持人：梁婷
共同主持人：
計畫參與人員： 吳典松 朱俊榮 施並格 林裕祥 黃立泓
施曉茹 蘇傳堯

成果報告類型(依經費核定清單規定繳交)：□精簡報告 ˇ完整報告

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
ˇ出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、列管計畫及下列情形者外，得立即公開查詢
□涉及專利或其他智慧財產權，□一年□二年後可公開查詢

執行單位：國立交通大學資訊工程系

中 華 民 國 95 年 9 月 13 日

資訊萃取技術在生物醫學文獻上的應用與探討

中文摘要

在本計畫中我們將探討兩個議題分別是萃取技術的研發和問答系統的製作將分兩年來進行。在第一年我們開發有效實用的自然語言處理技術和文件探勘技術，進而建製一個可應用在生物文獻的自動資訊萃取系統。主要的工作將包括生物實體名稱辨識、名稱指代處理、關係的辨識與萃取。我們結合法則式和統計式的方法來強化實體名稱辨識的效能。此外我們利用文件探勘技術來解決語句中指式型指代間題。同時我們也探討生物訊息和非生物訊息在實體關係的辨識和強度計算上的影響力，並利用探勘技術建立關聯法則以處理存在於語句中的實体關係的語言問題。

在第二年我們利用所開發的萃取技術進行以生物資訊為內容的知識問答糸統的製作，主要的工作將包括生物資料庫的內容探勘分類、查詢問題的分類、答案的選取和整合。

我們希望藉由此計劃的執行，一方面能開發出有效可行的資訊萃取方法將大量的生物文獻資料轉換成加值型的知識庫；另一方面亦提供使用者一個有效的知識萃取與處理系統,以促進生物資訊的探勘。

關鍵詞：自然語言處理、資訊萃取、文件探勘、實體名稱、指代處理、關係辨識、問答系統

Information Extraction In Biomedical Domain

Abstract

In this project, the issues associated with information extraction in biomedical domain are addressed in two years. In the first year, we develop an efficient information extraction system useful for biomedical literature by using natural language processing and textual mining techniques. This system will mainly address the tasks such as named entity identification, anaphora resolution, relation identification and extraction. We employ both statistical and linguistic models for named entities identification. We use textual mining to deal with those nominal anaphora problems. Meanwhile, the proposed relation recognition mechanism takes into account both the biomedical information encoded in the existing databases as well as the information directly mined from the literature. Besides, the problems associated with the linguistic varieties are tackled by using the proposed association rules.

In the second year we develop an on-line biomedical question answering system by applying information extraction techniques. The system addresses the issues such as question assembling and analysis, passage retrieval and answer extraction. In the proposed system answers can be extracted from corpus as well as semi-structured databases through different mining techniques. It is expected that the constructed system will be useful for the tasks such as knowledge acquisition and annotation.

We believe that the implementation of this project will be benefit for the tasks for knowledge acquisition and management, and, furthermore, potential scientific discovery.

Keywords: natural language processing, textual mining, information extraction, named entity identification, anaphora resolution, relation identification, question answering.

Table of Content

1 Introduction

In this project, the techniques useful for information extraction from biomedical literature are explored. The techniques involve the tasks like named entity recognition, anaphora recognition and biomedical relation recognition. Besides, a prototype of question answering system in biomedical domain is implemented with the application of these proposed techniques.

Named entity recognition (NER) from biomedical literature is one fundamental task involved in the automation of biomedical databases. Similar to the recognition in general domains, the issues associated with biomedical entity recognition are open vocabulary, synonyms, boundaries and sense disambiguation. In this project, both empirical rule and statistical approaches to protein entity recognition are presented and investigated on a general corpus GENIA 3.02p and a new domain-specific corpus SRC. Experimental results show the rules derived from SRC are useful though they are simpler and more general than the one used by other rule based approaches. Meanwhile, a concise HMM-based model with rich set of features is presented and proved to be robust and competitive while comparing it to other successful hybrid models. Besides, the resolution of coordination variants common in entities recognition is addressed. By applying heuristic rules and clustering strategy, the presented resolver is proved to be feasible.

As to the anaphora resolution in biomedical literature, it is noticed that pronominal and nominal anaphora are the two common types of anaphora. In this project, a resolution approach is presented by using rich set of syntactic and semantic features. Unlike previous researches, the verification of semantic association between anaphors and their antecedents is facilitated by exploiting more outer resources, including UMLS, WordNet, GENIA Corpus 3.02p and PubMed. Moreover, the resolution is implemented with a genetic algorithm on its feature selection. Experimental results on different biomedical corpora showed that such approach could achieve promising results on resolving the two common types of anaphora.

For relation recognition from biomedical literature, the complex sentence analysis presented in past literature is not practical enough to deal with rapid growth of biomedical literature. Some researchers using patterns to extract relation have been presented, yet, for example, the relations between two proteins locating at different sentences are not considered. In order to enhance the recognition accuracy, more features are considered in this project. A two-stage method for extracting protein-protein interactions from biomedical literature is proposed. In the first stage, patterns are utilized to match sentences containing interaction relation. In the second stage, a Naïve Bayes classifier is constructed by considering more features, like surface features, co-occurrence, co-citations, and protein property features. We use two corpora as our testing data. One is collected from MEDLINE abstracts, containing 155 abstracts, and the other containing 100 abstracts is collected from the references for proving interactions in DIP. We use the interaction pairs from DIP to justify our extraction method. The result shows that our approach can yield 62% and 61% F-score in both corpora, respectively.

At last, we implemented a prototype of specific-domain question answering. As we know, automation of question answering task involves question processing, information retrieval and answer extraction. It is noticed that more than 60% of QA errors are attributed to question processing. Hence the presented QA approach is designed with the aim to enhance QA performance by concerning question type classification and query expansion. Generally, more explanation questions are raised by a user using a system like medical QA system. The questions are like “Who is at the greatest risk for heat-related illness?” rather than “Who invented the toothbrush?” Hence, the proposed system is constructed with the exploitation of outer ontologies like UMLS and a domain-specific search engine like PubMed. Unlike most

previous researches focusing on UMLS as the domain expansion, we use the concepts in UMLS to extract Concept-Verb-Concept patterns (“CVC patterns” for short) from training corpus so as to improve the rank of answer texts. We use Naïve bayes model for question analysis so as to classify questions into diagnosis, therapy, and etiology and use query expansion to increase the recall for document retrieval. A combined ranking is presented for ranking answer texts and it is proved to yield promising results on 203 questions in terms of 0.63 MRR.

2 Related Works

2.1 Related works on biomedical NER

Recent textual mining approaches useful to biomedical NER can be divided into rule-based, statistical and hybrid methods. Generally, rule-based approaches employ the information of terms and hand-craft rules to produce candidates which are then verified by using lexical analysis [1] [2] [5]. Yet rule-based methods are essentially lack of portability and scalability. On the other hand, statistical models have been widely employed for their portability and scalability, such as Hidden Markov Model (HMM), Support Vector Model (SVM), Maximum Entropy (ME), and etc. The recognition accuracy achieved by these models generally depends on a well-tagged training corpus and a well set of features [3] [6] [7] [8] [9]. Recently, hybrid approaches are proposed by combining coded rules, statistical model and dictionaries [4] [8]. As pointed in [9], it can be expected that systems on a specified evaluation corpus with help of dictionaries tend to perform better than the general ones without help of any dictionaries. For example, the recognition performance is significantly improved when both dictionary and rules are applied together with a ME-based recognition mechanism in [4].

2.2 Related works on anaphora resolution in biomedical literature

In past literature, different strategies for resolving anaphora have been presented by using syntactic, semantic and pragmatic clues. For example, grammatical roles of noun phrases were used in [14] [15]. In addition to the syntactic information, statistical information like co-occurring patterns obtained from a corpus is employed during antecedent finding in [11]. However, a large corpus is needed for acquiring sufficient co-occurring patterns and for dealing with data sparseness. On the other hand, outer resources, like WordNet, are applied in [12] [17] [18] and proved to be helpful to improve the performance of an anaphora resolution system like the one presented in [17] where animacy information is exploited by analyzing the hierarchical relation of nouns and verbs in the surrounding context learned from WordNet. Nevertheless, using WordNet alone for acquiring semantic information is not sufficient for solving unknown words. To tackle this problem, a richer resource, the Web, was exploited in [19] where anaphoric information is mined from Google search results at the expense of less precision.

The domain-specific ontologies like UMLS (Unified Medical Language System) has been employed in [10] in such a way that frequent semantic types associated to agent (subject) and patient (object) role of subject-action or action-object patterns can be extracted. The result showed such kind of patterns could gain increase in both precision (76% to 80%) and recall (67% to 71%). On the other hand, Kim and Park [16] built their BioAR to relate protein names to SWISS-Prot entries by using the centering theory presented by [13] and salience measures by [10].

2.3 Related works on relation recognition in biomedical literature

There are several approaches presented for extracting relations from biomedical literature. For example, system GENIS [24] was designed to deal with a wide variety of different relations between biological molecules by analyzing most frequently used sentence structures. On the other hand, Daraselia et al. [22] utilized an ontology as a filter to select correct sentence structures and they have high precision 91% but low recall 21%. In order to improve the efficiency and reduce the workload of processors, some researchers use shallow parsers [20] [21] [23]. They identified certain phrases and extract dependencies between subject and object relationship without considering the structure of an entire sentence.

Unlike NLP techniques, researchers, like SUISEKI [28], employed a set of patterns which were predefined manually by filtering large amounts of text. These patterns are used to identify a direct or indirect interaction between two proteins. Huang et al. [25] used a dynamic programming algorithm to discover interaction patterns in the way of aligning relevant sentences and key verbs for identifying protein interactions. They extracted the interactions between proteins by matching the discovered patterns and the recall and precision rate were 80% and 80.5%, respectively. Oyama et al. [27] extracted the features that characterize each protein appearing in the interactions from several databases, like SWISS-PROT and PIR, and mined the association rules from interaction-based transactions. Ramani et al. [26] took an advantage of co-occurrence analysis to extract protein pairs from Medline abstracts.

2.4 Related works on question answering in biomedical domain

Many researches [29] [33] [35] [36] related to specific domain QA have been reported during the last decade. The specific domain QA is usually considered into four steps: the utilization of domain ontology, question processing, document retrieval, and answer processing. Zhang *et al.* [36] uses the concepts of ontology to tag the question and the documents in order to measure the similarity between the question and the documents. Wang *et al*. [35] consider the ontology as the keyword expansion for the question in order to gain more information. Soo *et al*. [33] integrate the biological literatures from the Web into the ontology automatically. The method presented in [29] uses the medical FAQ from the Web as the data source for the medical QA. In the this project, we consider how to utilize the concepts of ontology and the medical resources, i.e. medical FAQ and literatures, to deal with the medical questions in question processing and document retrieval.

For question processing, most specific domain QA adopts question classification as the essential component to deal with the given questions. Researches classify the questions by identifying the format of answers, such as Yes/No format [29], description format [29] [36], and NE format [36]. In our study, the concept information and the syntactic relation from the given question are concerned in order to make document retrieval work efficiently. A knowledge-based approach proposed by Navigl *et al.* [32] is used to do word sense disambiguation. Furthermore, the frequency of co-occurrence in UMLS is used to identify the concept.

For document retrieval, the okapi function is used to score the question concepts and keywords for retrieving the documents [36]. On the other hand, query expansion will increase the performance for document retrieval [30]. So the relations in the UMLS Metathesaurus are used to expand the query in [29] which the hierarchical relations are concerned as the important clue to increase the performance in document retrieval.

For answering definitional questions, Xu *et al.* [31] consider the linguistic features as the important clues to extract the definitions from the documents. With the growth of Web, the surface patterns [34] are utilized to collect the definitions from Web. In the project, we use the

definition database from UMLS to answer the definitional question. If the definition is not found in it, the online dictionary is queried to answer the question and expand the definition database at the same time.

3 The Proposed Methods and Results

3.1 The proposed named entities recognition

In this project, the recognition for protein entities from PubMed corpus is addressed so as to facilitate the automation of protein interaction databases construction. In order to mine more features relevant to protein entities, we assembled a domain-specific protein corpus SRC (SwissProt_Ref Corpus) extracted from SwissProt reference articles and tagged it by using SRC entry collection. The kernel NER is approached with two empirical strategies. One is rule-based strategy which exploits the patterns information mined from SRC. Experimental results show that the derived patterns are useful for NER task even though the number of the patterns is relatively less than the rules used in two popular systems Kex or Yapex. On the other hand, a concise HMM-based strategy is presented with a back-off strategy to overcome data sparseness. Experimental results on both GENIA corpus and the domain-specific SRC showed that the presented approach could achieve promising results in terms of 77% F-score in the case of strict annotation, proving that our approach is portable and competitive.

Besides, the recognition of the entities in coordination variants is concerned in this project. To resolve such term variants, a method based on heuristic rules together with clustering strategies is presented. Experimental results on GENIA corpus 3.0 proved the feasibility of the proposed approach by achieving 88.51% recall and 57.04% precision.

For detail description about the proposed method, please refer the attached conference paper presented in NLDB 2005, Alicante, Spain.

3.2 The proposed anaphora resolution for biomedical literature

In this project, a resolution procedure as shown in Figure 3.1 is presented for tackling both nominal anaphora and pronominal anaphora in biomedical literature by using morphological, syntactic and semantic clues. For nominal anaphora resolution, semantic association between anaphora and its antecedents is predicted with the semantic lexicons mined from UMLS and WordNet. For unknown entities, the semantic association is discovered by mining the search results with the help of PubMed, the search engine for MEDLINE databases. On the other hand, semantic coercion type of pronominal anaphor is done by semantic-tagged SA/AO patterns, which were pre-collected from GENIA 3.02p corpus. Unlike manual decision of feature sets at salience grading on antecedent selection, the presented resolution is boosted with a genetic algorithm. Experimental results on the evaluation corpus MedStract, the presented resolution is promising for its 92% F-Score in pronominal anaphora and 78% F-Score in nominal anaphora.

For detail description about the proposed method, please refer the attached conference paper presented in IJCNLP 2005, Jesu Island, South Korea.

3.3 The proposed relation recognition from biomedical literature

In this project, the interactions between protein pairs are addressed. The SWISS-PROT database is used as our lexicon to identify protein entities in corpus by maximum matching procedure. Through corpus preprocessing, protein pairs are formed and processed by the proposed extraction method. As shown in Figure 3.2, the proposed relation extraction is

divided into two stages. In the first stage, a set of predefined patterns mined from training corpus is employed to recognize relations from the testing sentences. In the second stage, the classifier based on Naive Bayes model is used for classifying each protein pair into two classes: “yes” or “no” by using a rich set of features which are verified with the Chi-Square test. The predefined features are described in detail in TABLE 1.

In order to select the best features, we incorporate the presented classifier with a genetic algorithm. TABLE 2 shows that we can have 74% F-score with the selected features and it is indeed better than the results yielded by using all features. TABLE 3 shows the impact of each feature in the training data. It reveals that the reference similarity feature plays a critical role for interaction extraction. Besides, the recognition performance is also justified with two corpora “Corpus1” and “Corpus2” with the best set of features selected by the genetic algorithm. (‘Corpus1 contains 155 Medline abstracts, and “Corpus2” contains 100 abstracts collected from the references listed in DIP.) The experiment results are displayed in TABLE 4 and TABLE 5, respectively. We can find that 61% F-score is achieved on both corpora, showing that the two-stage method is feasible for relation extraction.

For detail description about the proposed method, please refer the master thesis done by Hsiao-Ju Shih, Institute of Computer Science and Engineering, National Chiao Tung University 2006.

3.4 The proposed specific-domain question answering

The proposed QA processing is shown in Fig. 3.3 in which a given question is first identified to be is definitional or not. If the question is definitional type, the definitional strategy will be involved to process the question. If the question is the other types, a Naïve-Bayes classifier is employed to classify the questions into three target types. On the other hand, we use ontology-based expansion to expand the query term in order to increase the recall. Finally, we measure the returned texts by considering both TF-IDF and extracted concept patterns. Details of the implementation steps are described in the remaining subsections.

3.4.1 Rule-based approach for identifying definitional question

There are 108 definitional questions which have been classified manually in 910 pairs of the collected FAQs. We parse these questions and analyze the sentence structure. There are 88% definitional questions parsed as the following two structures.

Grammar: [Question Word + Be + Noun Phrase]
Question Word: What | Who
Be: is | are | was | were | be
Noun Phrase: ((Term1) (Term2) (Term3)…headword) |
((Term1 (Term2 (Term3 (…)))) headword)

The headword is the most important word for the noun phrase in the parsing tree. And then we can take the noun phrase to search the definitions in UMLS. The rules used to recognize definitional questions are listed as follows:

(i). The length of POS sequence is less and equal than four.
(ii). ["What or Who" + "be" + NP], the question structure is identified as structure 1 or structure 2.
(iii). The question contains only one NP.
(iv). There are no prepositions in NP.

In the experiment, we take 40 definitional questions from TREC-9 to evaluate the definitional rules. The experimental results show that 36 questions are detected by these rules. The accuracy rate is 90% in the test data. Some errors are resulted from wrong parsing tree or tags.

3.4.2 Naïve-Bayes classifier for classifying other type questions

A Naïve-Bayes classifier is used to classify the non-definitional questions into the pre-defined types, namely: diagnosis, therapy and etiology. We collect 8,729 medical documents classified by PubMed as the training data. Then we filter out stop words or medical proper nouns in UMLS. The remaining monograms (single word) and bigrams (adjacent two words) are clustered into 18 groups by a typical K-means algorithm. Meanwhile, we extract POS sequence from the classified questions and use POS sequence as one feature for our classifier. We follow the Bayesian Theorem (defined by Equation (1)) to train the question classifier by the features of grams and POS sequence. Each question is assigned with one unique question type. In the testing phase, we take 453 questions randomly from the rest FAQs. There are 85% precision and 86% recall for diagnosis, 84% precision and 94% recall for therapy and 82% precision and 88% recall for etiology.

3.4.3 Concept identification

Concept identification is presented with the help of UMLS for each medical phrase in the question so as to transform the NP-Verb-NP pattern into CVC pattern. Since UMLS is the multi-node structure, it is necessary for us to do concept disambiguation. We use the co-occurrence information in UMLS and the concept probabilistic function is designed as equation (3). Then we use the association function defined as (2) to measure which concepts are the most possible one to be associated in the sentence. Details of concept identification steps are summarized as following.

Algorithm for Concept Identification

IF the question contains only one noun phrase
THEN we get all concepts for the noun phrase from UMLS
OTHERWISE
(i). Identify all concepts for noun phrases
(ii). Calculate the probability for all concepts of the noun phrases according to the co-occurrence in UMLS
(iii). Calculate the association value to choose the most possible concept by equation (2) and assign it to the noun phrase

$$Prob_c = \arg\max_c P(C)\prod_{k=1}^{3} P(F_i \mid C) \quad (1)$$

C = {diagnosis, therapy, etiology}
F_i = {unigram, bigram, POS sequence}

$$Association(X_r, Y_h) = Prob(X_r \rightarrow Y_h) * Prob(Y_h \rightarrow X_r) \quad (2)$$

$$Prob(X_r \rightarrow Y_h) = \frac{freq(X_r, Y_h)}{freq(X_r, *)} \quad (3)$$

$X_r \in \{X_1, X_2 \ldots, X_i\}$, $Y_h \in \{Y_1, Y_2 \ldots, Y_j\}$
$freq(X_r, *)$: any concepts in UMLS co-occur with concept X_r
$freq(X_r, Y_h)$: concept X_r co-occur with concept Y_h

The extracted CVC patterns are used to score the answer texts in information retrieval. In the training phase, we use 400 medical terms as the keywords in UMLS to query the PubMed and collect 8,729 medical abstracts for training materials. The strategy is that all noun phrase preceding and succeeding the key verbs are extracted in the medical abstracts. If the noun phrase is a pronoun, the noun phrase which is preceded or succeeded the pronoun is extracted instead of the pronoun. Then noun phrases are combined with their preceding and succeeding verb as NP-Verb-NP patterns which are then transformed into CVC patterns.

For the verb in CVC patterns, we use the synsets of verb in WordNet to cluster CVC patterns into 4,496 groups and then we weigh each CVC pattern by equation (4).

$$Degree(CVC_i) = \frac{freq(C_A, Verb, C_B)}{freq(C_A, Verb) + freq(Verb, C_B) - freq(C_A, Verb, C_B)} \quad (4)$$

freq(Verb,C_B) = the co-occurrence for (Verb,C_B)
freq(C_A,Verb) = the co-occurrence for (C_A,Verb)
freq(C_A,Verb,C_B) = the co-occurrence for (C_A,Verb,C_B)

At run time, we use CVC pattern extracted from the given question to retrieve the stored CVC patterns from the training result and use the relevant CVC patterns to score the answer texts returned by search engine.

3.4.4 Ontology-based Query Expansion

The query expansion is done with the the synonyms and hierarchical relations in UMLS Metathesaurus. The expanded strategy is described as follows:

For each medical term in query

(i). Add the synonym variants in UMLS to the query
(ii). Add its parent terms in UMLS to the query
(iii). Add its child terms in UMLS to the query
(iv). Add other relations defined in UMLS to the query

3.4.5 Retrieval procedure and ranking

In the proposed QA, we use PubMed as the major information retrieval platform and Google as the minor platform. PubMed is triggered to retrieve the relevant medical texts if there exists. If not, Google will be triggered to retrieve the snippets according to the keywords from the given question.

The answer texts are measured by equation (5) based on TF-IDF.

$$\sum W_{i,j} = \sum (0.5 + \frac{0.5\, freq_{i,j}}{\max\ freq_{i,j}}) * \log \frac{N}{n_i} \quad (5)$$

$freq_{i,j}$: the frequency of term i in the answer text j
N: the number of answer texts
$n_{i,j}$: the number of answer texts containing term i

Beside the TF-IDF rank, we also compute the rank for each CVC of the answer texts by scoring the degree of the CVC patterns checked in common between the question and the answer texts.

3.4.6 Results and Analysis

Two indicators are used to measure the performance for our method. One is the Mean Reciprocal Rank (MRR). Another is the Human Effort (HE). The HE is defined as the user finds the answer in the least rank of passages returned.

Table 6 shows the experimental results on 55 questions from testing corpus and it is noticed that the proposed question classification (QC), query expansion (QE) and CVC patterns ranking indeed improve the QA performance. Table 7 shows the experimental results on 203 set-aside FAQ questions of different types. Table 8 shows the experimental results on the questions from view point of interrogative words. Table 9 shows the results in terms of Human Effort (HE) and it shows that the answer passage is at the top 2 (or top 3) in the returned texts from the proposed QA.

There are some errors attributed to the following reasons:

(1) Incorrect POS tagging.

(2) Assign the wrong category for the given question.

(3) Assign the not appropriate concept to noun phrase.

For detail description about the proposed method, please refer the master thesis done by Li-Hong Huang, Institute of Computer Science and Engineering, National Chiao Tung University 2006.

4 Concluding Remarks

In this project, we presented different textual mining strategies and natural language techniques for resolving biomedical knowledge extraction from on-line biomedical literature. We address four basic issues, namely, entity recognition, anaphora resolution, relation recognition and question answering.

The proposed entity recognition is focused on protein entities in this project. Both empirical rule and statistical approaches to protein entity recognition are presented and investigated on a general corpus GENIA 3.02p and a new domain-specific corpus SRC. Experimental results show the rules derived from SRC are useful though they are simpler and more general than the one used by other rule based approaches. Meanwhile, a concise HMM-based model with rich set of features is presented and proved to be robust and competitive while comparing it to other successful hybrid models. Besides, recognition for the entities in coordination variants is also concerned. To our best knowledge, our approach is the first one to cope with the term variants in the named entity extraction from biomedical texts. Partial results of this research have been presented in NLDB2005, Alicante, Spain. (“*Empirical Textual Mining to Protein Entities Recognition from PubMed Corpus”*, NLDB 2005, Lecture Notes in Computer Science 3513, pp. 56-66, 2005. (SCI extended)).

The second issue is the resolution for pronominal and nominal anaphora in biomedical literature. The resolution is constructed with a salience grading on various kinds of syntactic and semantic features. Unlike previous researches, we exploit more resources including both domain-specific and general thesaurus and corpus while dealing with semantic and syntactic agreement between anaphors and their antecedents. Experimental results on different corpora

prove that the semantic features provided with the help of the outer resources indeed can enhance anaphora resolution. Compared to other approaches, the presented best-first strategy with the genetic-algorithm based feature selection can achieve the best resolution on the same evaluation corpus. Partial results of this research have been presented in IJCNLP 2005, Jesu Island, Korea. (*Anaphora Resolution for Biomedical Literature by Exploiting Multiple Resources*, IJCNLP 2005, Lecture Notes in Artificial Intelligence 3651, pp. 742-753, 2005. (SCI extended)).

The third issue is automation of relation recognition among entities. In this project, we focus the protein interaction recognition from biomedical literature by employing both database and textual mining techniques. Unlike previous researches which are generally based on linguistic methods, a two-stage recognition approach is proposed in this project with the aim to improve the recognition recall. The first stage involves utilizing linguistic patterns which imply interaction relation from sentence structures. The second stage is based on a Naïve Bayes classifier which employs a rich set of features, including surface features, co-occurrence, co-citations, and protein features. We use two corpora as our testing data. One is a corpus of 155 MEDLINE abstracts, and the other contains 100 abstracts which are collected from the references for proving interactions in DIP (Database of Interaction Proteins). The result shows that our approach can yield 62% and 61% F-score on both corpora and it indeed enhance the low recall yielded by a general linguistic recognition approach.

The fourth issue we addressed in this two-year project is the implementation of a specific-domain QA prototype which is able to efficiently resolve the questions frequently raised by end-users. We apply UMLS, a domain–specific ontology to query expansion. Beside, we present a new answer passage ranking by weighing the transformed concept patterns mined at the training phase. The patterns provide a more general outlook for medical QA with respect to different kinds of question types. The presented QA is verified with different kinds of questions by various measurements. The results show that the proposed QA is able to retrieve the answer passage in the top 2 (or top 3) returned texts. Partial results of this research have been presented in IJCNLP 2005, Jesu Island, Korea. (*"Web-based Unsupervised Learning for Query Formulation in Question Answering"*, IJCNLP 2005, Lecture Notes in Artificial Intelligence 3651, pp. 519-529, 2005. (SCI extended)).

.

5 Reference

[1] Fukuda, K., Tsunoda, T., Tamura, A., and Takagi, T.: Towards Information Extraction: identifying Protein Names from Biological Papers. The 3rd Pacific Symposium on Biocomputing. (1998) 707-718.

[2] Hou, W. J. and Chen, H. H.: Enhancing Performance of Protein Name Recognizers using Collocation. ACL 2003 Workshop on Natural Language Processing in Biomedicine, (2003) 25-32.

[3] Lee, K.J., Hwang, Y.S., and Rim, H.C.: Two-Phase Biomedical NE Recognition based on SVMs. ACL 2003 Workshop on Natural Language Processing in Biomedicine, (2003) 33-40.

[4] Lin, Y., Tsai, T., Chiou, W. Wu, K., Sung, T.-Y., and Hsu, W-L.: A Maximum Entropy Approach to Biomedical Named Entity Recognition. 4th Workshop on Data Mining in Bioinformatics (2004).

[5] Olsson, F., Eriksson, G., Franzen, K., Asker, L., and Liden, P.: Notions of Correctness when Evaluating Protein Name Taggers. 19th International Conference on Computational Linguistics. (2002) 765-771.

[6] Settles, B.: Biomedical Named Entity Recognition Using Conditional Random Fields and Rich Feature Sets. Int'l Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA), Geneva, Switzerland (2004).

[7] Takeuchi, K. and Collier, N.: Bio-Medical Entity Extraction using Support Vector Machines. ACL 2003 Workshop on Natural Language Processing in Biomedicine, (2003) 57-64.
[8] Zhou, G.D. and Su, J.: Named Entity Recognition using an HMM-based Chunk Tagger. 40th Annual Meeting of the Association for Computational Linguistics (2002).
[9] Zhou, G., Zhang, J., Su, J., Shen, D. and Tan, C. L.: Recognizing Names in Biomedical Texts: A Machine Learning Approach. Bioinformatics, Vol. 20, (2004)1178-1190.
[10] Castaño, J., Zhang J., Pustejovsky, H.: Anaphora Resolution in Biomedical Literature. In International Symposium on Reference Resolution (2002)
[11] Dagan, I., Itai, A.: Automatic processing of large corpora for the resolution of anaphora references. In Proceedings of the 13th International Conference on Computational Linguistics (COLING'90) Vol. III (1990) 1-3
[12] Denber, M.: Automatic resolution of anaphora in English. Technical report, Eastman Kodak Co. (1998)
[13] Grosz, B.J., Joshi, A.K., Weinstein, S.: Centering: A framework for modelling the local coherence of discourse. Computational Linguistics 203.225 (1995)
[14] Hobbs, J.: Pronoun resolution, Research Report 76-1. Department of Computer Science, City College, City University of New York, August (1976)
[15] Kennedy, C., Boguraev, B.: Anaphora for everyone: Pronominal anaphora resolution without a parser. In Proceedings of the 16th International Conference on Computational Linguistics (1996) 113-118
[16] Kim, J., Jong, C.P.: BioAR: Anaphora Resolution for Relating Protein Names to Proteome Database Entries. ACL Workshop on Reference Resolution and its Applications Barcelona Spain (2004) 79-86
[17] Liang, T., Wu, D.S.: Automatic Pronominal Anaphora Resolution in English Texts. In Computational Linguistics and Chinese Language Processing Vol.9, No.1 (2004) 21-40
[18] Mitkov, R., Evans, R., Orasan, C.: A new fully automatic version of Mitkov's knowledge-poor pronoun resolution method. In Proceedings of CICLing- 2000 Mexico City Mexico (2002)
[19] Modjeska, Natalia, Markert, K., Nissim, M.: Using the Web in Machine Learning for Other-Anaphora Resolution. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP2003) Sapporo Japan
[20] G. Leroy and H. Chen, "Filling preposition-based templates to capture information from medical abstracts," in *Proc. 7th Pacific Symposium on Biocomputing*, pp.350-361, 2002.
[21] J. Pustejovsky, J. Castano, and J. Zhang, "Robust relational parsing over biomedical literature: extracting inhibit relations," in *Proc. 7th Pacific Symposium on Biocomputing*, pp.362-373, 2002.
[22] S. Novichkova, S. Egorov, and N. Daraselia, "MedScan, a natural language processing engine for MEDLINE abstracts," Bioinformatics, vol. 19, no. 13, pp. 1699-1706, 2003.
[23] G. Leroy, H. Chen, and J. D. Martinez, "A shallow parser based on closed-class words to capture relations in biomedical text," Journal of Biomedical Informatics, vol. 36, pp. 145-158, 2003.
[24] C. Friedman, P. Kra, H. Yu, M. Krauthammer, and A. Rzhetsky, "Genies: a natural-language processing system for the extraction of molecular pathways from journal articles," Bioinformatics, vol. 17, pp. 74-82, 2001.
[25] M. L. Huang, X. Y. Zhu, Y. Hao, D. G. Payan, K. B. Ou, and M. Li, "Discovering patterns to extract protein-protein interactions from full texts," Bioinformatics, vol. 20, no. 18, 2004.
[26] A. Ramani, E. Marcotte, R. Bunescu, and R. Mooney, "Using biomedical literature mining to consolidate the set of known human protein-protein interactions," in *Proc.*

ACL-ISMB Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics, pp. 46-53, 2005.
[27] T. Oyama, K. Kitano, K. Satou, and T. Ito, "Extraction of knowledge on protein-protein interaction by association rule discovery," Bioinformatics, vol. 18, no. 5, pp. 705-714, 2002.
[28] C. Blaschke and A. Valencia, "The frame-based module of the SUISEKI information extraction system," *IEEE* Intelligent Systems, pp. 14-20, 2002.
[29] C. H. Wu, J. F. Yeh, and M. J. Chen, "Domain-Specific FAQ Retrieval Using Independent Aspects," *ACM Transactions on Asian Language Information Processing*, Vol. 4, No. 1, pp. 1-17, March, 2005.
[30] D. Moldovan, M. Pasca, S. Harabagiu, and M. Surdeanu, "Performance Issues and Error Analysis in an Open-domain Question Answering System," *In Proceedings of ACM Transactions on Information Systems*, vol. 21, pp. 133-154, 2003.
[31] J. Xu, R. Weischedel, and A. Licuanan, "Evaluation of an Extraction-Based Approach to Answering Definitional Questions," *In Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval(SIGIR-2004)*, pp. 418 – 424, 2004.
[32] R. Navigli and P. Velardi, "Structural Semantic Interconnections: A Knowledge-Based Approach to Word Sense Disambiguation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 27, Issue 7, pp. 1075 - 1086, July, 2005.
[33] V. W. Soo, H. Y. Yeh, S. N. Lin, and W. C. Chen, "Ontology-based Knowledge Extraction from Semantic Annotated Biological Literatures," *The Ninth Conference on Artificial Intelligence and Applications*, 2004.
[34] W. Hildebrandt, B. Katz, and J. Lin, "Answering Definition Questions Using Multiple Knowledge Sources," *In Proceedings of the 2004 Human Language Technology Conference and the North American Chapter of the Association for Computational Linguistics Annual Meeting (HLT/NAACL 2004)*, Boston, Massachusetts, pp.49-56, 2004
[35] Y. C. Wang, J. C. Wu, T. Liang, and J. S. Chang, "Using the Web as Corpus for Un-supervised Learning in Question Answering," *In Proceedings of ROCLING*, pp.191-198, 2004.
[36] Z. Zhang, L. D. Sylva, C. Davidson, G. Lizarralde, and J. Y. Nie, "Domain-Specific QA for the Construction Sector," In Workshop of *ACM SIGIR Conference*, July 29, 2004.

Appendix

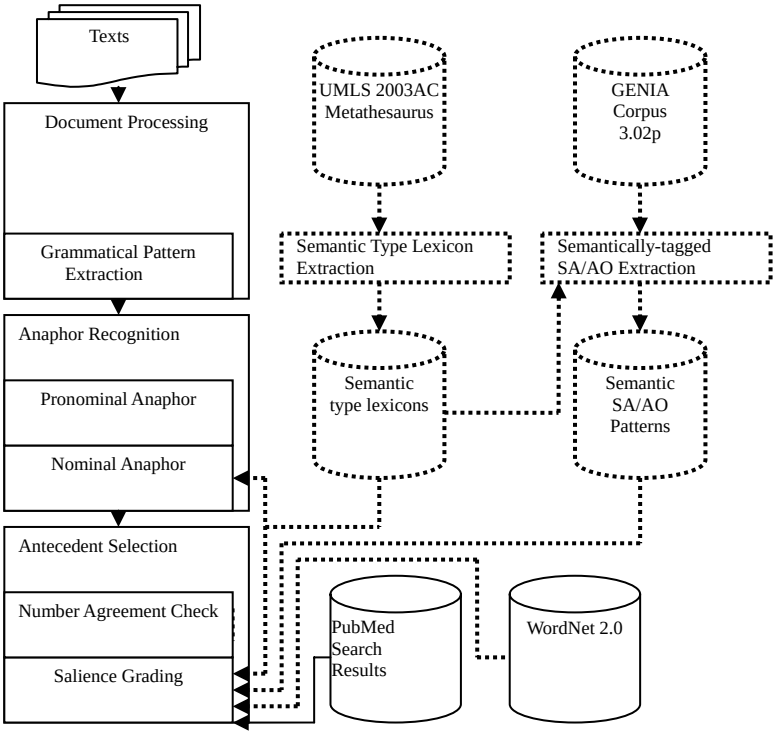

Fig. 1. System architecture overview.

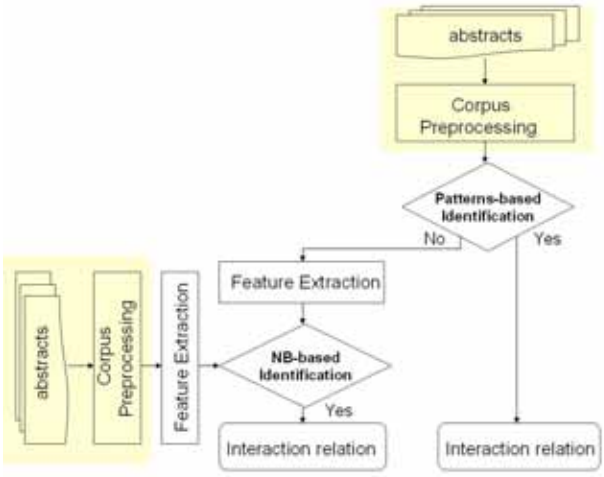

Fig. 2. Extraction flowchart.

TABLE 1

THE FEATURES DESCRIPTION

Feature	No	Description
Distance	1	The dice value of the frequencies of the protein pair in the same sentences
	2	The average of minimum distances of the protein pair in an abstract
Word	3	The cosine value of the protein pair which are presented as m-word vectors
Co-citation	4	The dice value of the frequencies of the protein pair in the same abstracts searched by the PUBMED.
	5	The maximum of reference similarities for protein pair.
Topic	6	The similarity of the topic “function” in the SwissProt database.
	7	The similarity of the topic “similarity” in the SwissProt database.
	8	The similarity of the topic “subcellular location” in the SwissProt database.
	9	The similarity of the topic “subunit” in the SwissProt database.
	10	The similarity of the topic “catalytic activity” in the SwissProt database.

TABLE 2

FEATURE SELECTION EXPERIMENTAL RESULTS WITH TRAINING CORPUS

Feature	Precision	Recall	F-score
Total features	68.91%	75.01%	71.83%
Genetic features all-{f8,f10}	72.55%	75.58%	74.49%

TABLE 3

THE FEATURE IMPACT ON THE TRAINING DATA

	features	precision	recall	F-score	Diff.
Training Set	All	68.91%	75.01%	71.83%	
	All-f1	65.77%	74.68%	69.94%	-1.89%
	All-f2	65.99%	73.77%	69.67%	-2.17%
	All-f3	65.10%	76.48%	70.33%	-1.50%
	All-f4	68.86%	71.46%	70.14%	-1.70%
	All-f5	59.81%	60.52%	60.16%	-11.67%
	All-f6	65.61%	74.79%	69.90%	-1.93%
	All-f7	67.61%	75.24%	71.22%	-0.61%
	All-f8	69.54%	75.47%	72.38%	0.55%
	All-f9	65.44%	74.45%	69.66%	-2.18%
	All-f10	70.15%	73.04%	71.57%	-0.27%

TABLE 4

RELATION IDENTIFICATION RESULTS ON TEST CORPUS1

	Precision	Recall	F-score
First Stage	61.11%	30.56%	40.74%
Second Stage	51.06%	57.60%	54.13%
Total	54.98%	70.56%	61.80%

TABLE 5

RELATION IDENTIFICATION RESULTS ON TEST CORPUS2

	Precision	Recall	F-score
First Stage	61.11%	30.56%	40.74%
Second Stage	51.06%	57.60%	54.13%
Total	54.98%	70.56%	61.80%

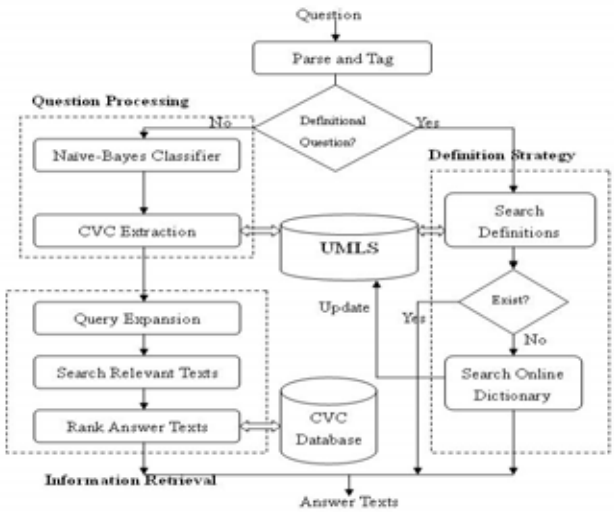

Fig. 3 Flowchart of QA processing

Table 6
MRR of each component

	MRR
QC+QE+CVC	0.63
QC+QE	0.58
QC+CVC	0.57

Table 7. MRR for different type question

	Number of Questions	MRR
Diagnosis	103	0.62
Therapy	45	0.67
Etiology	55	0.62

Table 8
MRR for the interrogative words

	What	When	Who	Where	Why	How
Number of Questions	78	8	13	11	5	88
MRR	0.63	0.54	0.65	0.64	0.66	0.64

Table 9
Human effort for each component

Rank	Rank Count			
	Diagnosis	Therapy	Etiology	All Types
1	48	24	27	99
2	19	9	6	34
3	9	3	6	18
4	3	0	2	5
5	3	0	3	6
No Answer	21	9	11	41
# of questions	103	45	55	203
HE per question	2.58	2.33	2.65	2.55

Three attached papers

Empirical Textual Mining to Protein Entities Recognition from PubMed Corpus

Tyne Liang and Ping-Ke Shih

Department of Computer and Information Science
National Chiao Tung University, Hsinchu, Taiwan
tliang@cis.nctu.edu.tw

Abstract. Named Entity Recognition (NER) from biomedical literature is crucial in biomedical knowledge base automation. In this paper, both empirical rule and statistical approaches to protein entity recognition are presented and investigated on a general corpus GENIA 3.02p and a new domain-specific corpus SRC. Experimental results show the rules derived from SRC are useful though they are simpler and more general than the one used by other rule-based approaches. Meanwhile, a concise HMM-based model with rich set of features is presented and proved to be robust and competitive while comparing it to other successful hybrid models. Besides, the resolution of coordination variants common in entities recognition is addressed. By applying heuristic rules and clustering strategy, the presented resolver is proved to be feasible.

1 Introduction

Nowadays efficient automation of biomedical knowledge bases is urgently demanded to cope with the proliferation of biomedical researches. One crucial task involved in the automation is named entity recognition (NER) from biomedical literature. Similar to the recognition in general domains, the issues associated with biomedical entity recognition are open vocabulary, synonyms, boundaries and sense disambiguation. For example, the number of entries in SwissProt[1], a protein knowledge base, increases 277.36% in recent ten years. Each protein entity contains 2.54 synonyms in average, and each synonym contains 2.74 tokens in average.

Recent textual mining approaches useful to biomedical NER can be divided into rule-based, statistical and hybrid methods. Generally, rule-based approaches employ the information of terms and hand-craft rules to produce candidates which are then verified by using lexical analysis [1, 2, 5]. Yet rule-based methods require more domain knowledge and essentially lack of scalability. On the other hand, statistical models have been widely employed for their portability and scalability, such as Hidden Markov Model (HMM), Support Vector Model (SVM), Maximum Entropy (ME), and etc.. The recognition accuracy achieved by these models generally depends on a well-tagged training corpus and a well set

[1] SwissProt: http://us.expasy.org/sprot/

A. Montoyo et al. (Eds.): NLDB 2005, LNCS 3513, pp. 56–66, 2005.

of features [3, 6, 7, 9, 10]. Recently, hybrid approaches are proposed by combining coded rules, statistical model and dictionaries [4, 9]. As pointed in [10], it is expected that systems on a specified evaluation corpus with help of dictionaries tend to perform better than the general ones without help of any dictionaries. For example, the recognition performance is significantly improved when dictionary and rules are applied at post-processing together with a ME-based recognition mechanism in [4].

In this paper, recognition for protein entities from PubMed[2] corpus is addressed so as to facilitate the automation of protein interaction databases construction. In order to mine more features relevant to protein entities, we assembled a domain-specific protein corpus SRC (SwissProt Reference Corpus) which were extracted from SwissProt reference articles and we tagged it by simply matching SwissProt entry collection. Experimental results show that this new domain corpus is indeed helpful in generating informative patterns used in both rule-based and statistical models. It is also found that though the derived rules are fewer and less complicated than the ones used in the rule-based systems Kex [1] or Yapex [5], the presented model outperforms these two systems in terms of higher F-scores on a general corpus like GENIA 3.02p [3] and the domain-specific SRC.

On the other hand, a concise HMM-based model is presented with a back-off strategy to overcome data sparseness. With a rich set of features, the presented approaches could achieve promising results, by showing 76-77% F-scores on both GENIA corpus and SRC. Compared to the results achieved by some successful systems (the best 78% F-score for protein instances in [9]) which employ dictionaries or semantic lexicon lists, our results are competitive for three reasons. First, the recognition is done without any help of dictionaries or predefined lexicon lists. Second, the presented concise HMM is easily implemented and robust for different corpora. Third, our results are evaluated with strict annotation and enetities with the longest annotation are adopted in case they are in the nested forms.

Besides, this paper addresses the issue of coordination variants while we tackle with NER problems in written texts. To resolve such term variants, a method based on heuristic rules and clustering strategy is presented. Experimental results on GENIA corpus 3.0 proved its feasibility by achieving 88.51% recall and 57.04% precision on a test of 1850 sentences, including 174 variants.

2 Corpus Preparation

In order to boost protein entities recognition by mining more relevant information, we assembled a domain-specific corpus 'SwissProt Ref Corpus' ('SRC' for short), other than the widely-used tagged corpus like GENIA 3.02p. The new corpus was processed by employing Sentence Splitter[4] and Penn Treebank

[2] PubMed: http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Pubmed
[3] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/
[4] Sentence Splitter: http://l2r.cs.uiuc.edu/~cogcomp/

Tokenizer[5] for sentence segmentation and tokenization respectively. The POS-tagging is processed by a HMM-based POS tagger which was developed in our lab. By using GENIA 3.02p as training set, our POS-tagger could yield 95% F-score. For the sake of saving human efforts, annotating SRC with all the target entities was simply implemented with the following steps:

1. Tokens are split by space and hyphen.
2. Each token is converted to lower case except its initial character.
3. Entity is recognized if it matches an entity from SwissProt version 42.0.

The final specific SRC corpus is composed of 2,894 abstracts, which were particularly selected from SWISSPORT 82,740 reference articles in such a way that each of them contains at least six target entities. Table 1 lists the basic statistics for SRC and GENIA 3.02p.

Table 1. The statistics of SRC corpus and GENIA corpus 3.02p.

	SRC		**GENIA**	
	count	average	count	average
Abstract (a)	2,894		1,999	
Sentence (s)	28,154	9.73 (s/a)	18,572	9.29 (s/a)
Token (t)	740,001	255.70 (s/a)	490.469	245.36 (t/a)
		26.28 (t/s)		26.41 (t/s)
Protein (p)	31,977	11.05 (p/a)	32,525	11.05 (p/a)
Entity		1.14 (p/s)		1.14 (p/a)
Entity Token (t)	57,878	1.81(t/p)	58,200	1.79 (t/p)

3 Coordination Variants Resolution

Coordination variants are one common type of variants in general written texts like MEDLINE records. For example there are 1598 coordination variants in GENIA 3.02p corpus and each variant contains 2.1 entities in average. Table 2lists three types of the regular expressions generalized from the GENIA 3.02p training corpus of 16,684 sentences (in which 1421 coordination variants are distributed in 1329 sentences). There #, H, T, and R indicate core, head, tail, and coordinate terms respectively. For example, in the coordination '91 and 84 kDa proteins', '91' and '84' are the core terms, 'kDa proteins' is the tail term, and 'and' is the coordinate term.

The variant resolution was implemented with finite state machines (FSM) which are verified by a test set of 1850 sentences in which 174 variants are distributed in 165 sentences. Experimental results showed that this approach yielded 91.38% recall and 42.06% precision (indicated as "baseline approach"in Table 3). In practice, the precision can be improved by presenting more number of FSMs so as to cover all possible variant patterns, yet it will slow down the resolving throughput. In order to increase the sensitivity of coordination identification, a simple term clustering is employed. Suppose terms t_i, t_j co-occur

[5] http://www.cis.upenn.edu/~treebank/tokenization.html

Table 2. Original patterns, expanded patterns, and examples.

	Regular Expression		Example
Type 1	Original	$H\#(R\#)^+$	human chromosomes 11p15 and 11p13
	Expanded	$(H\#R)^+H\#$	humman chromosomes 11p15 and human chromosome 11p13
Type 2	Original	$\#(R\#)^+T$	c-fos, c-jun, and EGR2 mRNA
	Expanded	$\#T(R\#)^+T$	c-fos mRNA, c-jun mRNA, and EGR2 mRNA
Type 3	Original	$H\#(R\#)^+T$	human T and B lymphocytes
	Expanded	$\#T(R\#)^+T$	human T lymphocytes and human B lymphocytes

in one coordination variant, and terms t_i, t_k co-occur in another one. Then we put t_i, t_j and t_k into one cluster. The clustering procedure was implemented recursively. With such term clustering strategy (indicated as 'unlimited-distance' in Table 3), the resolution precision is increased by 4%. This showed that the clustering approach is helpful to restrict the path movement in FSMs. To distinguish the closeness of the terms in the same cluster, we furthermore applied the Floyd-Warshall algorithm to cluster sets. That is, if terms t_i, t_j co-occur in a sentence and terms t_i, t_k co-occur in another one but t_j, t_k do not co-occur in any sentence, then the $dist(t_j, t_k) = 2$. With this clustering strategy, the precision became 57.04% (increasing 15% with respect to the baseline method) at the expense of lower recall.

Table 3. Accuracy of coordination variants identification in GENIA 3.02p.

	dist.	Variants	tp+fp	tp	Recall	Precision	F-Score
Baseline	N/A	174	378	159	91.38%	42.06%	57.61%
Term Clustering	unlimited	174	338	158	90.80%	46.75%	61.72%
	1	174	270	154	88.51%	57.04%	69.37%

4 Protein Entity Recognition

In this paper, protein entity recognition is approached and investigated by both rule-based and HMM models. The performance verification is implemented by using both SRC and GENIA 3.02p corpora in such a way that the corpora are divided into 90% for training phase and 10% for testing phase.

4.1 Rule-Based Approach

The rule-based recognition is implemented by employing the patterns of the protein nomenclature mined from SRC and GENIA corpora. The patterns are formed in terms of core, function or predefined terms. Core terms show the closest resemblance to regular proper names. Function terms describe the functions or characteristics of a protein. Table 4 shows the frequent regular expressions which 'C' indicates core term, 'F' indicates function term, and 'P' indicates predefined term, namely specifier, amino acid and unit.

Table 4. Top 5 regular expressions of protein entities in SRC and GENIA 3.02p.

Regular Expression	SRC	Regular Expression	GENIA
C^+	25.70%	C^+	69.64%
C^+F^+	21.22%	C^+F^+	8.14%
F^+	15.57%	C^+P^+	5.84%
F^+P^+	12.62%	F^+C^+	2.91%
C^+P^+	9.36%	F^+	2.35%

The function terms may be head or tail function term depending on the position they appear texts. From our observation of SRC, 58.48% head function terms appear before an initial uppercase token, and 74.07% tail function terms appear after an initial uppercase token or a specifier. We define 217 head function terms and 127 tail function terms. The rest of the terms other than predefined and function terms are treated as core terms candidates. The candidates may be the composition of common strings which are useful for identifying unknown words. For example, a common string 'CD' is acquired from a core term 'CD23', and then an unknown word 'CD25' will be seen as a core term.

The extraction of protein entities is done by six steps. The first three steps are aimed to produce the candidates by using term information. If a token is one of the three type terms, it will be annotated. Steps 4-6 are aimed to acquire protein entities as many as possible.

Step 1: boundary confirmation We scan the chunk forward (left to right) and backward (right to left) to fix entity boundaries by exploiting POS pattern information of protein entities, as shown in Tables 5 and 6.

Table 5. Top 5 POS patterns in SRC and GENIA.

POS Pattern	SRC	POS Pattern	GENIA
NN	79.38%	NN	67.57%
NN,CD	12.94%	JJ,NN	7.13%
JJ,NN	3.13%	NNS	7.11%
JJ,NN	3.02%	JJ,NNS	2.94%
CD,NN	0.26%	NN,CD	0.96%

Table 6. The top frequent POS tags at the first and the last positions of chunks.

	First POS tag		Last POS tag	
POS	SRC	GENIA	SRC	GENIA
CD	0.27%	0.43%	13.12%	1.91%
JJ	6.32%	13.23%	3.03%	0.57%
NN	93.12%	83.20%	83.43%	83.50%
NNS	0.01%	2.28%	0.08%	13.66%
VBN	0.14%	0.31%	0.08%	0.01%

Step 2: remove invalid single-token chunks A single-token chunk will be treated as invalid if (a) its characters are in lower case, and the token is not a protein entity in training data or (b) it is a predefined term only.

Step 3: remove invalid multi-token chunks by using a general set of domain-independent rules. A chunk will be removed if it composes of the followings: (a) the predefined terms, (b) the single uppercase English letters, (c) the punctuation marks, and (d) the conjunctions. After the three steps, 68.21% and 52.63% invalid tokens in SRC and GENIA are removed 98.58% and 96.93% accuracy rates respectively.

Step 4: mine the tokens surrounding protein entities This step is to acquire more protein entities. The pattern is formulated as '$< T_{-2}, T_{-1}, \#, T_1, T_2 >$', where '#' is token's number of the protein entity, and the token 'T_i' is the i^{th} token relative to the protein entity. Two measurements namely, confidence and occurrence are used to justify the usefulness of the patterns. Confidence is the ratio of the number of correct instances divided by the number of all instances in training data, and occurrence is the number of all instances in training data. Patterns are selected whenever their occurrence and confidence are greater than one and 0.8 respectively, because our system is expected to achieve 80% correct rate, which is the ratio of the number of correct instances divided by the number of all retrieved instances.

Step 5: mine the bag-of-word surrounding protein entities For each protein entity we collect its preceding two tokens and following two tokens. The non-confidence is used to filter the candidates and it is defined as the ratio of the negative instances to all instances. Patterns are recognized whenever non-confidence is greater than 0.8 since our system is expected to yield 80% correct rate.

Step 6: employ syntactic rules Hypernyms may appear in front of hyponyms, and one common pattern is 'NP_0 such as $\{NP_1, NP_2, \ldots,$ (and|or) $\}$ NP_n'. So we can mine those clue words by collecting the tokens preceding 'such as' and 'e.g.'. For example, 'protein' is the clue token of '... proteins, such as CBL and VAV, were phosphorylated on ...'. The clue words are the tokens of UMLS concepts and their corresponding synonyms which are tagged with 'protein' semantic type.

The model performance is evaluated in terms of precision (P), recall (R) and F-score (F) which is 2PR/(R+P). To present performance of rule-based systems, we use the notations of correct matching defined in [5]. Table 7 shows that the strict measure, which the proposed hit matches one answer key exactly, can yield 51%-52% F-Score. Table 7 shows that we can get higher F-score if we measure the performance with PNP ('protein name parts'), meaning each proposed token matches any token of the answer key. For example 'CD surface receptor' is treated as 'PNP' of 'activation of the CD28 surface receptor'. In practice, such kind of annotation result is acceptable. In addition, Table 7 also shows that the terms, mined from SRC, are adaptable since we can obtain almost the same performance results from GENIA corpus. Table 8 shows the improvement is obvious for steps 1 to 3, but steps 4 to 6 have little effect. On the other hand, the precision can be boosted obviously but not much for recall.

Table 7. Experimental results by rule-based approach.

	Notation	tp+sn	tp+fp	tp	recall	precision	F-Score
SRC	SLOPPY	3234	4782	2987	92.36%	62.46%	74.53%
	PNP	3234	4782	2859	88.40%	59.79%	71.33%
	STRICT	3234	4782	2077	64.22%	43.43%	51.82%
	LEFT	3234	4782	2620	81.01%	54.79%	65.37%
	RIGHT	3234	4782	2363	73.07%	49.41%	58.96%
	LorR	3234	4782	2907	89.89%	60.79%	72.53%
	Notation	tp+sn	tp+fp	tp	recall	precision	F-Score
GENIA	SLOPPY	3451	4923	3010	87.22%	61.14%	71.89%
	PNP	3451	4923	2837	82.21%	57.63%	67.76%
	STRICT	3451	4923	2123	61.52%	43.12%	50.70%
	LEFT	3451	4923	2765	80.12%	56.16%	66.04%
	RIGHT	3451	4923	2296	66.53%	46.64%	54.84%
	LorR	3451	4923	2938	85.13%	59.68%	70.17%

Table 8. The intermediate results of rule-based approach.

	Procedure	tp+sn	tp+fp	tp	recall	precision	F-Score
SRC	step1	3234	10480	2051	63.42%	19.57%	29.91%
	step1-2	3234	5493	2043	63.17%	37.19%	46.82%
	step1-3	3234	4911	2040	63.08%	41.54%	50.09%
	step1-4	3234	4977	2104	65.06%	42.27%	51.25%
	step1-5	3234	4781	2077	64.22%	43.33%	51.83%
	step1-6	3234	4782	2077	64.22%	43.43%	51.82%
	Procedure	tp+sn	tp+fp	tp	recall	precision	F-Score
GENIA	step1	3451	7911	2160	62.59%	27.30%	38.02%
	step1-2	3451	5173	2129	61.69%	41.16%	49.37%
	step1-3	3451	5082	2127	61.63%	41.85%	49.85%
	step1-4	3451	5164	2155	62.45%	41.73%	50.03%
	step1-5	3451	4915	2120	61.43%	43.13%	50.68%
	step1-6	3451	4923	2123	51.52%	43.12%	50.70%

4.2 HMM-Based Approaches

The statistical approach for NER is implemented by a concise HMM model (Concise-HMM) which employs a rich set of input features. Its performance is verified with SRC and GENIA 3.02p by comparing two other models, namely, traditional model (Traditional-HMM) and mutual information model (MI-HMM) which was presented in [9] and produced high F-scores in MUC-6 and MUC-7. The comparison is made in the same environment settings.

In this paper, all the models are trained with the same set of useful features including internal, external and global features. Internal features are those surface clues in tokens (e.g. initial character is upper case). There are 17 internal features mined from the training corpus. External features indicate the external information associated with tokens. We treated POS tags as our external features. Global features are the trigger nouns extracted from whole training

corpus by using Chi-square test. Besides, the complete-link clustering algorithm is applied to the mined nouns so as to reduce their dimensions. For window size of three sentences, we have 214 and 142 noun clusters in SRC and GENIA corpus respectively.

Traditional HMM. Given a token sequence $T_1^n = t_1 t_2 \dots t_n$, the goal is to find an optimal state sequence $S_1^n = s_1 s_2 \dots s_n$ that maximizes $\log Pr(S_1^n|T_1^n)$, the logarithm probability of state sequence S_1^n corresponding to the given token sequence T_1^n. By applying Bayes's rule to

$$Pr(S_1^n|T_1^n) = \frac{Pr(S_1^n|T_1^n)}{Pr(T_1^n)} \tag{1}$$

we have

$$\arg\max_S \log Pr(S_1^n|T_1^n) = \arg\max_S \log Pr(S_1^n|T_1^n) + \log Pr(S_1^n)) \tag{2}$$

where

$$Pr(T_1^n|S_1^n) = \prod_{i=1}^{n} Pr(t_i|s_i) \tag{3}$$

and

$$Pr(S_1^n) = \prod_{i=1}^{n} Pr(s_i|s_{i-1}) \tag{4}$$

with the assumption of conditional probability independence and considering preceding state. Therefore equation (2) can be rewritten as:

$$\arg\max_S \log Pr(S_1^n|T_1^n) = \arg\max_S \left(\sum_{i-1}^{n} (\log Pr(t_i|s_i) + \log Pr(s_i|s_{i-1})) \right) \tag{5}$$

MI-HMM. Different from traditional HMM, MI-HMM is aimed to maximize the equation:

$$\arg\max_S \log Pr(S_1^n|T_1^n) = \arg\max_S \left(\log Pr(S_1^n) + \log \frac{Pr(S_1^n, T_1^n)}{Pr(S_1^n) \bullet Pr(T_1^n)}) \right) \tag{6}$$

In order to simplify the computation, the mutual information independence is assumed to be:

$$MI(S_1^n, T_1^n) = \sum_{i=1}^{n} MI(s_i, T_1^n) \tag{7}$$

or

$$\log \frac{Pr(S_1^n, T_1^n)}{Pr(S_1^n) \bullet Pr(T_1^n)} = \sum_{i=1}^{n} \log \frac{Pr(s_i, T_1^n)}{Pr(s_i) \bullet Pr(T_1^n)} \tag{8}$$

Applying it to equation (6), we have:

$$\arg\max_S \log Pr(S_1^n|T_1^n) = \arg\max_S \left(\log Pr(S_1^n) - \sum_{i=1}^{n} \log Pr(s_i) + \sum_{i=1}^{n} \log Pr(s_i|T_1^n) \right) \tag{9}$$

Concise HMM. The presented concise HMM is based on the idea of maximizing the fundamental $\log Pr(S_1^n|T_1^n)$. In the equation (9), $\log Pr(S_1^n|T_1^n)$ and $\sum_{i=1}^{n} \log Pr(s_i)$ are found to carry less meaning because the weak probabilities of states and state transitions are merely 3-by-3 and 3-by-1 matrices respectively. Thus, a concise HMM can be obtained by simplifying the formula (9) to be equation (10):

$$\arg \max_S \log Pr(S_1^n|T_1^n) = \arg \max_S \log Pr(S_1^n) - \sum_{i=1}^{n} \log Pr(s_i|T_1^n) \qquad (10)$$

Since the concise HMM does not take its state transition into account, we put previous state in the model to ensure correct state induction. Because the presented HMM approach concerned many features mentioned above, it is possible to train a high-accuracy probability model. To overcome spareseness problem, we use a back-off strategy which aims at the token sequence T_1^n in $Pr(S_1^n|T_1^n)$ or in $Pr(s_i|T_1^n)$ where T_1^n represents not only a token sequence but also the full set of sequence's features. There are two back-off levels. First level is based on different combinations of tokens and their features, and T_1^n will be assigned in the descending order:

$< s_{-1}, t_{-1}, t_0, f_0 >, < s_{-1}, t_0, f_0 >, < s_{-1}, t_{-1}, f_0 >, < s_{-1}, f_0 >$

where f_i represents the feature set including internal, external and global features. t_i is a token, s_i expresses a HMM state, and i is the i^{th} one relative to current token. Second level is based on different combinations of features, and f_i in first level is assigned in the descending order:

$< f_i^I, f_i^E, f_i^G >, < f_i^I, f_i^E >, < f_i^I >$

where f_i^I , f_i^E and f_i^G represent internal, external and global features, respectively.

4.3 Method Comparisons

Method comparisons for the three HMM-based models were made on both SRC corpus and GENIA corpus in the same environment settings. We used the same back-off model for concise and mutual information HMM, but not for traditional HMM. Table 9 shows that concise HMM with rule-based features (i.e. concise-ruled) yielded the best result. Traditional HMM obtains good high precision, but low recall since we chose a severe probability model to get the best F-score. It is also noticed that the performance of MI-HMM turned out to be the worst because the back-off model was used to optimize concise HMM. On the other hand, Table 10 shows all kinds of features turned out to be positive effect ($f^E > f^I > f^G$) for concise HMM. Such result is similar to that concluded from [10]. Table 11 lists the comparisons of the presented approaches to other well-known approaches on the public evaluation GENIA 3.x corpus. It is noticed that the presented rule-based approach with its simple general rules outperformed the other two complicated rule-based systems. On the other hand, the performance of the presented concise HMM-based models is comparable to the best model presented in [4]. However, we do not need any dictionary or rules in our model.

Table 9. HMM-based model comparison.

	HMM	tp+sn	tp+fp	tp	recall	precision	F-Score
SRC	Concise	3234	2953	2355	72.82%	79.75%	76.13%
	Concise-ruled	3234	2949	2391	73.93%	81.08%	77.34%
	MI	3234	3439	2384	73.72%	69.32%	71.45%
	Traditional	3234	2396	2086	64.50%	87.06%	74.10%
	HMM	tp+sn	tp+fp	tp	recall	precision	F-Score
GENIA	Concise	3451	3285	2553	73.98%	77.72%	75.80%
	Concise-ruled	3451	3323	2596	75.22%	78.12%	76.65%
	MI	3451	3415	2305	66.79%	67.50%	67.14%
	Traditional	3451	2863	2263	65.58%	79.04%	71.68%

Table 10. The effects of features in concise HMM.

	Features	tp+sn	tp+fp	tp	recall	precision	F-Score	Diff.
SRC	All	3234	2953	2355	72.82%	79.75%	76.13%	
	All-f^G	3234	2951	2335	72.20%	79.13%	75.51%	-0.62%
	All-f^E	3234	2894	2284	70.62%	78.92%	74.54%	-1.59%
	All-f^I	3234	2941	2303	71.21%	78.31%	74.59%	-1.54%
	Features	tp+sn	tp+fp	tp	recall	precision	F-Score	Diff.
GENIA	ALL	3451	3285	2553	73.98%	77.72%	75.80%	
	All-f^G	3451	3267	2534	73.43%	77.56%	75.44%	-0.36%
	All-f^E	3451	3176	2442	70.76%	76.89%	73.70%	-2.10%
	All-f^I	3451	3213	2467	71.49%	76.78%	74.04%	-1.76%

Table 11. Comparison to other systems on GENIA corpus.

System	Method	GENIA	Recall	Precision	F-Score
Lee et. al. [3]	SVM	3.0p	78.80%	61.70%	69.20%
Lin et. al. [4]	ME-hybrid	3.01	77.00%	80.00%	78.50%
KeX	Rule-based	3.02p	43.67%	37.40%	40.29%
Yapex	Rule-based	3.02p	45.06%	54.17%	47.48%
Ours	Rule-based	3.02p	61.52%	43.12%	50.70%
	concise-HMM	3.02p	73.98%	77.72%	75.80%
	concise-ruled	3.02p	75.22%	78.12%	76.64%

5 Conclusions and Future Work

In this paper, we presented different textual mining strategies applicable to supporting full automation of protein entities recognition. Recognition for the entities in coordination variants is also concerned. To our best knowledge, our approach is the first one to cope with the term variants in the named entity extraction from biomedical texts. On the other hand, practical textual mining to protein entities recognition were presented by both rule and statistical models. Without the help of any dictionaries, the kernel recognition based on a concise HMM-based model turns out to be promising for protein entity extraction.

Future work includes the manual annotation correction of SRC for fine classification, exploitation of dictionaries for better recognition performance and the improvement of the resolution for coordination variants by using the semantic type information of biomedical thesaurus like UMLS. In addition, novel mining techniques to resolve other types of term variants should be explored for full NER automation.

Acknowledgements

This research is partially supported by MediaTek Research Center, National Chiao Tung University, Taiwan and partially supported by National Science Council under the contract NSC 93-2213-E-009-074.

References

1. Fukuda, K. and Tsunoda, T. and Tamura, A. and Takagi, T.: Towards Information Extraction: identifying Protein Names from Biological Papers. The 3rd Pacific Symposium on Biocomputing. (1998) 707-718.
2. Hou, W. J. and Chen, H. H.: Enhancing Performance of Protein Name Recognizers using Collocation. ACL 2003 Workshop on Natural Language Processing in Biomedicine, (2003) 25-32.
3. Lee, K.J. and Hwang, Y.S. and Rim, H.C.: Two-Phase Biomedical NE Recognition based on SVMs. ACL 2003 Workshop on Natural Language Processing in Biomedicine, (2003) 33-40.
4. Lin, Y. and Tsai, T. and Chiou, W. and Wu K. and Sung, T.-Y. and Hsu, W-L.: A Maximum Entropy Approach to Biomedical Named Entity Recognition. 4th Workshop on Data Mining in Bioinformatics (2004).
5. Olsson, F., Eriksson, G., Franzen, K., Asker, L., and Liden, P.: Notions of Correctness when Evaluating Protein Name Taggers. 19th International Conference on Computational Linguistics. (2002) 765-771.
6. Settles, B.: Biomedical Named Entity Recognition Using Conditional Random Fields and Rich Feature Sets. Int'l Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA), Geneva, Switzerland (2004).
7. Takeuchi, K. and Collier, N.: Bio-Medical Entity Extraction using Support Vector Machines. ACL 2003 Workshop on Natural Language Processing in Biomedicine, (2003) 57-64.
8. Tsuruoka, Y. and Tsujii, J.: Boosting Precision and Recall of Dictionary-based Protein Name Recognition. ACL 2003 Workshop on Natural Language Processing in Biomedicine (2003) 41-48.
9. Zhou, G.D. and Su, J.: Named Entity Recognition using an HMM-based Chunk Tagger. 40th Annual Meeting of the Association for Computational Linguistics (2002).
10. Zhou, G., Zhang, J., Su, J., Shen, D. and Tan, C. L.: Recognizing Names in Biomedical Texts: A Machine Learning Approach. Bioinformatics, Vol. 20, (2004)1178-1190.

Anaphora Resolution for Biomedical Literature by Exploiting Multiple Resources

Tyne Liang and Yu-Hsiang Lin

National Chiao Tung University, Department of Computer and Information Science, Hsinchu, Taiwan 300, ROC
{tliang, gis91534}@cis.nctu.edu.tw

Abstract. In this paper, a resolution system is presented to tackle nominal and pronominal anaphora in biomedical literature by using rich set of syntactic and semantic features. Unlike previous researches, the verification of semantic association between anaphors and their antecedents is facilitated by exploiting more outer resources, including UMLS, WordNet, GENIA Corpus 3.02p and PubMed. Moreover, the resolution is implemented with a genetic algorithm on its feature selection. Experimental results on different biomedical corpora showed that such approach could achieve promising results on resolving the two common types of anaphora.

1 Introduction

Correct identification of antecedents for an anaphor is essential in message understanding systems as well as knowledge acquisition systems. For example, efficient anaphora resolution is needed to enhance protein interaction extraction from biomedical literature by mining more protein entity instances which are represented with pronouns or general concepts.

In biomedical literature, pronominal and nominal anaphora are the two common types of anaphora. In past literature, different strategies to identify antecedents of an anaphor have been presented by using syntactic, semantic and pragmatic clues. For example, grammatical roles of noun phrases were used in [9] [10]. In addition to the syntactic information, statistical information like co-occurring patterns obtained from a corpus is employed during antecedent finding in [3]. However, a large corpus is needed for acquiring sufficient co-occurring patterns and for dealing with data sparseness.

On the other hand, outer resources, like WordNet[1], are applied in [4][12][15] and proved to be helpful to improve the system like the one described in [12] where animacy information is exploited by analyzing the hierarchical relation of nouns and verbs in the surrounding context learned from WordNet. Nevertheless, using WordNet alone for acquiring semantic information is not sufficient for solving unknown words. To tackle this problem, a richer resource, the Web, was exploited in [16]

[1] http://wordnet.princeton.edu/

R. Dale et al. (Eds.): IJCNLP 2005, LNAI 3651, pp. 742 – 753, 2005.

where anaphoric information is mined from Google search results at the expense of less precision.

The domain-specific ontologies like UMLS[2] (Unified Medical Language System) has been employed in [2] in such a way that frequent semantic types associated to agent (subject) and patient (object) role of subject-action or action-object patterns can be extracted. The result showed such kind of patterns could gain increase in both precision (76% to 80%) and recall (67% to 71%). On the other hand, Kim and Park [11] built their BioAR to relate protein names to SWISS-Prot entries by using the centering theory presented by [7] and salience measures by [2].

In this paper, a resolution system is presented for tackling both nominal anaphora and pronominal anaphora in biomedical literature by using various kinds of syntactic and semantic features. Unlike previous approaches, our verification of the semantic association between anaphors and their antecedents is facilitated with the help of both general domain and domain-specific resources. For example, the semantic type checking for resolving nominal anaphora can be done by the domain ontology UMLS and PubMed[3], the search engine for MEDLINE databases. Here, UMLS is used not only for tagging the semantic type for the noun phrase chunks if they are in UMLS, but also for generating the key lexicons for each type so that we can use them to tag those chunks if they are not in UMLS. If no type information can be obtained from an chunk, then its type finding will be implemented through the web mining of PubMed. On the other hand, the domain corpus, GENIA 3.02p corpus [20] is exploited while we solve the semantic type checking for pronominal anaphora. With simple weight calculation, the key SA/AO (subject-action or action-object) patterns for each type can be mined from the corpus and they turn out to be helpful in resolution. Beside the semantic type agreement, the implicit resemblance between an anaphor and its antecedents is another evidence useful for verifying the semantic association. Hence, the general domain thesaurus, WordNet, which supporting more relationship between concepts and subconcepts, is also employed to enhance the resemblance extraction.

The presented resolution system is constructed on a basis of a salience grading. In order to boost the system, we implemented a simple genetic algorithm on its selection of the rich feature set. The system was developed on the small evaluation corpus MedStract[4]. Nevertheless, we constructed a larger test corpus (denoted as '100-MEDLINE') so that more instances of anaphors can be resolved. Experimental results show that our resolution on MedStract can yield 92% and 78% F-Scores on resolving pronominal and nominal anaphora respectively. Promising results were also obtained on the larger corpus in terms of 87.43% and 80.61% F-scores on resloving pronominal and nominal anaphora respectively.

2 Anaphora Resolution

Figure 1 is the overview of the presented architecture, including the extraction of biomedical SA/AO patterns and semantic type lexicons in background processing

[2] http://www.nlm.nih.gov/research/umls/
[3] http://www.pubmedcentral.nih.gov/
[4] http://www.medstract.org/

(indicated with dotted lines), as well as the document processing, anaphor recognition and antecedent selection in foreground processing (indicated with solid lines).

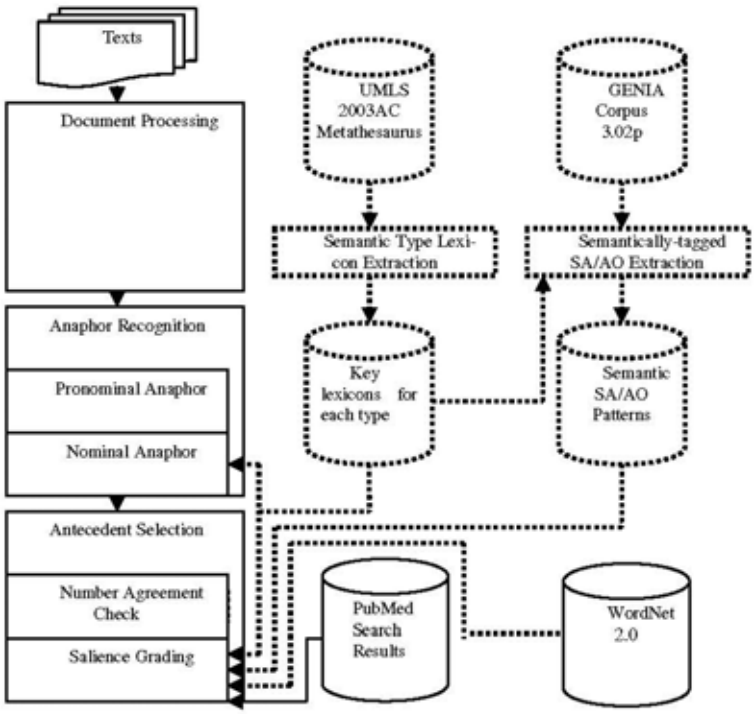

Fig. 1. System architecture overview

2.1 Syntactic Information Extraction

Being important features for anaphora resolution, syntactic information, like POS tags and base NP chunks, is extracted from each document by using the Tagger[5]. Meanwhile, each NP will be tagged with its grammatical role, namely, 'Oblique', 'Direct object', 'Indirect object', or 'Subject' by using the following rules which were adopted from [22] by adding rules 5 and 6.

Rule1:	Prep NP (Oblique)
Rule2:	Verb NP (Direct object)
Rule3:	Verb [NP]* NP (Indirect object)
Rule4:	NP (Subject) [",[^Verb], "|Prep NP]* Verb
Rule5:	NP1 Conjunction NP2 (Role is same as NP1) Conjunction]
Rule6:	[Conjunction] NP1 (Role is same as NP2) Conjunction NP2

[5] http://tamas.nlm.nih.gov/tagger.html

Rules 5 and 6 are presented for dealing those plural anaphors in such a way that the syntactic agreement between the first antecedent and its anaphora is used to find other antecedents. For example, without rules 5 and 6, 'anti-CD4 mAb' in Example 1 will not be found when resolving anaphora 'they'.

> Example1: "Whereas different anti-CD4 mAb or HIV-1 gp120 could all trigger activation of the ..., they differed…"

2.2 Semantic Information Extraction

Beside the syntactic clues, the semantic agreement between an anaphor and its antecedents can also facilitate anaphora resolution in domain-specific literature. In this paper, the semantic information for each target noun phrase chunk can be extracted with the help of the domain ontology, UMLS, which supports the semantic type for the chunk. However, the semantic types for those chunks which are not in UMLS are needed to be predicted. Therefore we need to extract the key lexicons from UMLS for each semantic type in background processing and use them to tag unknown chunk with predicted types. On the other hand, the semantic type checking for pronominal anaphors is done through the extraction of the key verbs for each semantic type. Hence, a domain corpus GENIA 3.02p is exploited in background processing.

2.2.1 Key Lexicons for Each Semantic Type

For each UMLS semantic type, its key lexicons are mined as the following steps in Figure 2:

A. Collect all UMLS concepts and their corresponding synonyms as type lexicon candidates.
B. Tokenize the candidates. For example, concept 'interleukin-2' has synonyms 'Costimulator', 'Co-Simulator', 'IL 2', and 'interleukine 2'. Then 'interleukin', 'costimulator', 'simulator', 'IL', and 'interleukine' will be treated as lexicon candidates.
C. For each candidate, calculate its weight w_{ij} for each type by using Eq. (1) which takes into account its concentration and distribution. A predefined threshold is given for the final selection of the candidates.

$$w_{i,j} = \frac{w_i}{Max_{c_j}} \times \frac{1}{tw_i} \quad (1)$$

$w_{i,j}$: score of word i in semantic type j
w_i : count of word i in semantic type j
Max c_j : Max count of word k in semantic type j
tw_i : count of semantic types that word i occurs in

Fig. 2. Procedure to mine key lexicons for each semantic type

2.2.2 Semantic SA/AO Patterns

As indicated previously in Section 2.2, the semantic type checking for pronominal anaphors can be done through the extraction of the co-occurring SA/AO patterns extracted from GENIA 3.02p. We tagged each base noun phrase chunk from the corpus with its grammatical role and tagged it with UMLS-semantic type. Then we used Eq. 2 to score each pattern. At resolution, an antecedent candidate is concerned if its scores are greater than a given threshold. Table 1 is an example to show the key lexicons and verbs for two semantic types when the semantically-typed chunk is tagged with the role of subject.

$$score(type_i, verb_j) = \frac{frequency(type_i, verb_j)}{frequency(verb_j)} \times \frac{1}{No.\ of\ types(verb_j)} \quad (2)$$

Table 1. Some key lexicons and verbs for two semantic types

Semantic types	key lexicons for each type	key verbs for each type
Amino Acid, Peptide, or Protein	protein, product, cerevisiae, endonuclease, kinase, antigen, receptor, synthase, reductase, arabidopsis	bind, function, derive, raise, attenuate, abolish, present, signal, localize, release
Gene or Genome	gene, oncogenes	activate, compare, locate, regulate, remain, transcribe, encode, distribute, indicate, occupy

2.3 Anaphora Recognition

Anaphor recognition is to recognize the target anaphors by filtering strategies. Pronominal anaphora recognition is done by filtering pleonastic-it instances by using the set of hand-craft rules presented in [12]. On two corpora, namely, Medstract and the new 100-Medline corpus, 100% recognition accuracy was achieved. The remaining noun phrases indicated with 'it', 'its', 'itself', 'they', 'them', 'themselves' or 'their' are considered as pronominal anaphor. Others like 'which' and 'that' used in relative clauses are treated as pronominal anaphors and are resolved by the following rules.

Rule 1: 'that' is treated as pleonastic-that if it is paired with pleonastic-it.
Rule 2: For a relative clause with 'which' or 'that', the antecedents will be the noun phrases preceding to 'which' or 'that'.

On the other hand, noun phrases shown with 'either', 'this', 'both', 'these', 'the', and 'each' are considered as nominal anaphor candidates. Nominal anaphora recognition is approached by filtering those anaphor candidates, which have no referent antecedents or which have antecedents but not in the target biomedical semantic types. Following are two rules used to filter out those non-target nominal anaphors.

Rule 1: Filter out those anaphor candidates if they are not tagged with one of the target UMLS semantic types (the same types in [2])
Rule 2: Filter out 'this' or 'the' + proper nouns with capital letters or numbers.

We treated all other anaphors indicated with 'this' or 'the + singular-NP' as singular anaphors which have one antecedent only. Others are treated as plural nominal anaphors and their numbers of antecedents are shown in Table 2. At antecedent selection, we can discard those candidates whose numbers differ from the corresponding anaphors.

Table 2. Number of Antecedents

Anaphor	Antecedents #
Either	2
Both	2
Each	Many
They, Their, Them, Themselves	Many
The +Number+ noun	Number
Those +Number+ noun	Number
These +Number+ noun	Number

2.4 Antecedent Selection

2.4.1 Salience Grading

The antecedent selection is based on the salience grading as shown in Table 3 in which seven features, including syntactic and semantic information, are concerned.

Table 3. Salience grading for candidate antecedents

Features		Score
F1	recency 0, if in two sentences away from anaphor 1, if in one sentence away from anaphor 2, if in same sentence as anaphor	0-2
F2	Subject and Object Preference	1
F3	Grammatical function agreement	1
F4	Number Agreement	1
F5	Semantic Longest Common Subsequence	0 to 3
F6	Semantic Type Agreement	-1 to +2
F7	Biomedical antecedent preference	-2 if not or +2

The first feature *F1* is recency which measures the distance between an anaphor and candidate antecedents in number of sentences. From the statistics of the two corpora, most of antecedents and their corresponding anaphors are within in two sentence distance, so a window size for finding antecedent candidates is set to be two sentences in the proposed system. The second feature *F2* concerns the grammatical roles that an

anaphor plays in a sentence. Since many anaphors are subjects or objects so antecedents with such grammatical tags are preferred. Furthermore, the antecedent candidates will receive more scores if they have grammatical roles (feature *F3*) or number agreement (feature *F4*) with their anaphors.

On the other hand, features *5, 6,* and *7* are related to semantic association. Feature *5* concerns the fact that the anaphor and its antecedents are semantical variants of each other, so antecedents will receive different scores (as shown below) on the basis of their variation:

If there is total match of the semantic lexicons between an antecedent's head word and its anaphor
  Then salience score = salience score + 3
  Else If any antecedent component, other than head word, is matched with its anaphor
    Then salience score = salience score + 2
    Else If any antecedent component is matched with its anaphor's hyponym by WordNet 2.0
      Then salience score = salience score + 1

Following are examples to show the cases:

Example 2
case 1: total match:
  <anaphor: each *inhibitor*, antecedent: PAH alkyne metabolism-based *inhibitors*>
case 2: partial match:
  <Anaphor: both *receptor types*, antecedent: the ETB *receptor* antagonist BQ788>
case 3: component match by using WordNet 2.0:
  <Anaphor: this protein (hyponym: growth *factor*), antecedent: Cleavage and polyadenylation specificity *factor*>

If the antecedent can be found by UMLS,
  Then record its semantic types;
  Else If the antecedent contains the mined key lexicons of the anaphor's semantic type, then record the semantic type;
    Else mine the semantic type by web mining in such a way that searching PubMed by issuing {anaphor *Ana*, antecedent A_i } pair and applying Eq. 3 to grade its semantic agreement for A_i.

$$Score(A_i) = Score(A_i) - 1 + \left\lceil \frac{\#\ of\ pages\ containing(Ana, A_i)}{\#\ of\ pages\ containing(A_i)} \times 10 \right\rceil \times 0.3 \qquad (3)$$

Fig. 3. Procedure to find semantic types for antecedent candidates

Feature 6 is the semantic type agreement between anaphors and antecedents. As described in figure 3, the type finding for each antecedent can implemented with the help of UMLS. When there is no type information can be obtained from an antecedent, the type finding can be implemented with the help of PubMed, and the grading on such antecedent will be as Eq. 3. Feature 7 is biomedical antecedent preference. That is an antecedent which can be tagged with UMLS or the key lexicons database will receive more score.

2.4.2 Antecedent Selection Strategies

The noun phrases which precede a recognized anaphor in the range of two sentences will be treated as candidates and will be assigned with zero at initial state by the presented salience grader. Antecedents can be selected by the following strategies.

(1) Best First: select antecedents with the highest salience score that is greater than a threshold
(2) Nearest First: select the nearest antecedents whose salience value is greater than a given threshold

For plural anaphors, their antecedents are selected as follows:

(1) If the number of the antecedents is known, then select the same number of top-score antecedents.
(2) If the number of antecedents is unknown, then select those antecedent candidates whose scores are greater than a threshold and whose grammatical patterns are the same as the top-score candidate.

2.5 Experiments and Analysis

As mentioned in previous sections, a larger corpus was used for testing the proposed system. The corpus, denoted as '100-Medline', contains 100 MEDLINE abstracts including 43 abstracts (denoted as '43-Genia' in Table 6) randomly selected from GENIA 3.02p and another 57 abstracts (denoted as '57-PubMed' in Table 6) collected from the search results of PubMed (by issuing 'these proteins' and 'these receptors' in order to acquire more anaphor instances). There is no common abstract in the public MedStract and the new corpus. Table 4 shows the statistics of pronominal and nominal anaphors for each corpus.

Table 4. Statistics of anaphor and antecedent pairs

	Abstracts	Sentences	Pronominal instances	Nominal instances	Total
MedStract	32	268	26	47	73
43-GENIA	43	479	98	63	161
57-PubMed	57	565	69	118	187

The proposed approach was verified with experiments in two ways. One is to investigate the impact of the features which are concerned in the resolution. Another is to compare different resolution approaches. In order to boost our system, a simple

generic algorithm is implemented to yield the best set of features by choosing best parents to produce offspring.

In the initial state, we chose features (10 chromosomes), and chose crossover feature to produce offspring randomly. We calculated mutations for each feature in each chromosome, and evaluated chromosome with maximal F-Score. Top 10 chromosomes were chosen for next generation and the algorithm terminated if two contiguous generations did not increase the F-score. The time complexity associated with such approach is *O(MN)* where *M* is the number of candidate antecedents, *N* is number of anaphors.

Table 5. F-Score of Medstract and 100-Medlines

	Medstract						100-Medlines					
	Nominal			Pronominal			Nominal			Pronominal		
	P	R	F	P	R	F	P	R	F	P	R	F
Total Features	33/56	33/47		23/26	23/26		130/184	130/178		145/167	145/167	
	58.93	70.21	64.08	88.46	88.46	88.46	70.65	73.34	71.33	86.82	86.82	86.82
	F5, F6, F7			All-F5			F5, F6, F7			All-F5		
	P	R	F	P	R	F	P	R	F	P	R	F
Genetic Features	37/47	37/47		24/26	24/26		156/212	156/178		146/167	146/167	
	78.72	78.72	78.72	92.31	92.31	92.31	73.58	87.64	80.61	87.43	87.43	87.43

Table 6. Feature impact experiments

	Medstract		43-GENIA		57-PubMed	
	Nominal	Pronominal	Nominal	Pronominal	Nominal	Pronominal
All	64.08%	88.46%	67.69%	93.58%	73.28%	76.81%
All – F1	61.05%	73.08%	60.14%	83.87%	75.44%	75.36%
All – F2	65.96%	88.00%	70.22%	93.58%	78.40%	76.81%
All – F3	72.00%	80.77%	69.68%	84.46%	73.45%	76.81%
All – F4	64.65%	81.48%	68.33%	91.54%	73.73%	76.81%
All – F5	48.00%	92.31%	52.55%	93.58%	56.59%	78.26%
All – F6	44.04%	88.46%	46.42%	81.63%	57.14%	78.26%
All – F7	38.26%	59.26%	47.19%	71.96%	60.44%	50.72%

Table 5 shows that anaphora resolution implemented with the genetic algorithm indeed achieves higher F-scores than the one when all features are concerned. Table 5 also shows that the semantic features play more important role than the syntactic features for nominal anaphora resolution. Similar results can be also found in Table 6 where the impact of each feature is justified. Moreover, Table 6 indicates that the pronominal anaphora resolution on 43-Genia is better than that on the other two corpora. It implies that the mined SA/AO patterns from GENIA 3.02p corpus are

helpful for pronominal anaphora resolution. Moreover, Table 7 proves that the key lexicons mined from UMLS for semantic type finding indeed enhance anaphora resolution, yet a slight improvement is found with the usage of PubMed search results. One of the reasons is few unknown instances in our corpora.

On the other hand, comparisons with evaluation corpus, Medstract, were shown in Table 8 where the best-first strategy yielded higher F-score than the results by the nearest-first strategy. It also shows that the best-first strategy with the best selection by genetic approach achieves higher F-scores than the approach presented in [2].

Table 7. Impacts of the mined semantic lexicons and the use of PubMed

	With semantic lexicons		w/o semantic lexicons	
	Medstract.	100-Medlines	Medstract.	100-Medlines
With PubMed	78%	80.62%	59%	72.16%
Without PubMed	76%	80.13%	58%	71.33%

Table 8. Comparisons among different strategies on Medstract

	Best-First		Nearest-First		Castaño et al. [2]	
F-score	Nominal	Pronominal	Nominal	Pronominal	Nominal	Pronominal
Total Features	64.08%	88.46%	50.49%	73.47%		
Genetic Features	F5, F6, F7 78.72%	All - F5 92.31%	F5, F6, F7 61.18%	All-(F2,F5) 79.17%	F4, F5, F6 74.40%	F4, F6, F7 75.23%

3 Conclusion

In this paper, the resolution for pronominal and nominal anaphora in biomedical literature is addressed. The resolution is constructed with a salience grading on various kinds of syntactic and semantic features. Unlike previous researches, we exploit more resources, including both domain-specific and general thesaurus and corpus, to verify the semantic association between anaphors and their antecedents. Experimental results on different corpora prove that the semantic features provided with the help of the outer resources indeed can enhance anaphora resolution. Compared to other approaches, the presented best-first strategy with the genetic-algorithm based feature selection can achieve the best resolution on the same evaluation corpus.

References

1. Baldwin, B.: CogNIAC: high precision coreference with limited knowledge and linguistic resources. In Proceedings of the ACL'97/EACL'97 workshop on Operational factors in practical, robust anaphora resolution (1997) 38-45
2. Castaño, J., Zhang J., Pustejovsky, H.: Anaphora Resolution in Biomedical Literature. In International Symposium on Reference Resolution (2002)

3. Dagan, I., Itai, A.: Automatic processing of large corpora for the resolution of anaphora references. In Proceedings of the 13th International Conference on Computational Linguistics (COLING'90) Vol. III (1990) 1-3
4. Denber, M.: Automatic resolution of anaphora in English. Technical report, Eastman Kodak Co. (1998)
5. Gaizauskas, R., Demetriou, G., Artymiuk, P.J., Willett, P.: Protein Structures and Information Extraction from Biological Texts: The PASTA System. Bioinformatics (2003)
6. Gasperin, C., Vieira R.: Using word similarity lists for resolving indirect anaphora. In ACL Workshop on Reference Resolution and its Applications, Barcelona (2004)
7. Grosz, B.J., Joshi, A.K., Weinstein, S.: Centering: A framework for modelling the local coherence of discourse. Computational Linguistics (1995) 203-225
8. Hahn, U., Romacker, M.: Creating Knowledge Repositories from Biomedical Reports:The MEDSYNDIKATE Text Mining System. In Pacific Symposium on Biocomputing (2002)
9. Hobbs, J.: Pronoun resolution, Research Report 76-1. Department of Computer Science, City College, City University of New York, August (1976)
10. Kennedy, C., Boguraev, B.: Anaphora for everyone: Pronominal anaphora resolution without a parser. In Proceedings of the 16th International Conference on Computational Linguistics (1996) 113-118
11. Kim, J., Jong, C.P.: BioAR: Anaphora Resolution for Relating Protein Names to Proteome Database Entries. ACL Workshop on Reference Resolution and its Applications Barcelona Spain (2004) 79-86
12. Liang, T., Wu, D.S.: Automatic Pronominal Anaphora Resolution in English Texts. Computational Linguistics and Chinese Language Processing Vol.9, No.1 (2004) 21-40
13. Mitkov, R.: Robust pronoun resolution with limited knowledge. In Proceedings of the 18th International Conference on Computational Linguistics (COLING'98)/ACL'98 Conference Montreal Canada (1998) 869-875
14. Mitkov, R.: Anaphora Resolution: The State of the Art. Working paper (Based on the COLING'98/ACL'98 tutorial on anaphora resolution) (1999)
15. Mitkov, R., Evans, R., Orasan, C.: A new fully automatic version of Mitkov's knowledge-poor pronoun resolution method. In Proceedings of CICLing- 2000 Mexico City Mexico (2002)
16. Modjeska, Natalia, Markert, K., Nissim, M.: Using the Web in Machine Learning for Other-Anaphora Resolution. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP2003) Sapporo Japan
17. Navarretta, C.: An Algorithm for Resolving Individual and Abstract Anaphora in Danish Texts and Dialogues. ACL Workshop on Reference Resolution and its Applications Barcelona, Spain (2004) 95-102
18. Ng, V., Cardie, C.: Improving Machine Learning Approaches to Coreference Resolution. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics (2002)
19. Oh, I.S., Lee, J.S., Moon, B.R.: Hybrid Genetic Algorithms for Feature Selection. IEEE Transactions on pattern analysis and machine Vol. 26. No. 11 (2004)
20. Ohta, T., Tateisi, Y., Kim, J.D., Lee, S.Z., Tsujii, J.: GENIA corpus: A Semantically Annotated Corpus in Molecular Biology Domain. In Proceedings of the ninth International Conference on Intelligent Systems for Molecular Biology (ISMB 2001) poster session (2001) 68

21. Pustejovsky, J., Rumshisky, A., Castaño, J.: Rerendering Semantic Ontologies: Automatic Extensions to UMLS through Corpus Analytics. LREC 2002 Workshop on Ontologies and Lexical Knowledge Bases (2002)
22. Siddharthan, A.: Resolving Pronouns Robustly: Plumbing the Depths of Shallowness. In Proceedings of the Workshop on Computational Treatments of Anaphora, 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2003) (2003) 7-14
23. Yang, X., Su, J., Zhou, G., Tan, C.L.: Improving Pronoun Resolution by Incorporating Coreferential Information of Candidates. In Proceedings of ACL 2004 (2004) 127-134

Web-Based Unsupervised Learning for Query Formulation in Question Answering

Yi-Chia Wang[1], Jian-Cheng Wu[2], Tyne Liang[1], and Jason S. Chang[2]

[1] Dep. of Computer and Information Science, National Chiao Tung University,
1001 Ta Hsueh Rd., Hsinchu, Taiwan 300, R.O.C.
rhyme.cis92g@nctu.edu.tw, tliang@cis.nctu.edu.tw
[2] Dep. of Computer Science, National Tsing Hua University,
101, Section 2 Kuang Fu Road, Hsinchu, Taiwan 300, R.O.C.
d928322@oz.nthu.edu.tw, jschang@cs.nthu.edu.tw

Abstract. Converting questions to effective queries is crucial to open-domain question answering systems. In this paper, we present a web-based unsupervised learning approach for transforming a given natural-language question to an effective query. The method involves querying a search engine for Web passages that contain the answer to the question, extracting patterns that characterize fine-grained classification for answers, and linking these patterns with n-grams in answer passages. Independent evaluation on a set of questions shows that the proposed approach outperforms a naive keyword-based approach in terms of mean reciprocal rank and human effort.

1 Introduction

An automated *question answering* (QA) system receives a user's natural-language question and returns exact answers by analyzing the question and consulting a large text collection [1, 2]. As Moldovan et al. [3] pointed out, over 60% of the QA errors can be attributed to ineffective question processing, including query formulation and query expansion.

A naive solution to query formulation is using the keywords in an input question as the query to a search engine. However, it is possible that the keywords may not appear in those answer passages which contain answers to the given question. For example, submitting the keywords in "*Who invented washing machine?*" to a search engine like Google may not lead to retrieval of answer passages like "*The inventor of the automatic washer was John Chamberlain.*" In fact, by expanding the keyword set ("*invented*", "*washing*", "*machine*") with *"inventor of,"* the query to a search engine is effective in retrieving such answer passages as the top-ranking pages. Hence, if we can learn how to associate a set of questions (e.g. (*"who invented …?"*) with effective keywords or phrases (e.g. *"inventor of"*) which are likely to appear in answer passages, the search engine will have a better chance of retrieving pages containing the answer.

In this paper, we present a novel Web-based unsupervised learning approach to handling question analysis for QA systems. In our approach, training-data questions are first analyzed and classified into a set of fine-grained categories of question

R. Dale et al. (Eds.): IJCNLP 2005, LNAI 3651, pp. 519 – 529, 2005.

patterns. Then, the relationships between the question patterns and n-grams in answer passages are discovered by employing a word alignment technique. Finally, the best query transforms are derived by ranking the n-grams which are associated with a specific question pattern. At runtime, the keywords in a given question are extracted and the question is categorized. Then the keywords are expanded according the category of the question. The expanded query is the submitted to a search engine in order to bias the search engine to return passages that are more likely to contain answers to the question. Experimental results indicate the expanded query indeed outperforms the approach of directly using the keywords in the question.

2 Related Work

Recent work in Question Answering has attempted to convert the original input question into a query that is more likely to retrieve the answers. Hovy et al. [2] utilized WordNet hypernyms and synonyms to expand queries to increase recall. Hildebrandt et al. [4] looked up in a pre-compiled knowledge base and a dictionary to expand a definition question. However, blindly expanding a word using its synonyms or dictionary gloss may cause undesirable effects. Furthermore, it is difficult to determine which of many related word senses should be considered when expanding the query.

Radev et al. [5] proposed a probabilistic algorithm called *QASM* that learns the best query expansion from a natural language question. The query expansion takes the form of a series of operators, including INSERT, DELETE, REPLACE, etc., to paraphrase a factual question into the best search engine query by applying Expectation Maximization algorithm. On the other hand, Hermjakob et al. [6] described an experiment to observe and learn from human subjects who were given a question and asked to write queries which are most effective in retrieving the answer to the question. First, several randomly selected questions are given to users to "manually" generate effective queries that can bias Web search engines to return answers. The questions, queries, and search results are then examined to derive seven query reformulation techniques that can be used to produce queries similar to the ones issued by human subjects.

In a study closely related to our work, Agichtein et al. [7] presented *Tritus* system that automatically learns transforms of wh-phrases (e.g. expanding "*what is*" to "*refers to*") by using FAQ data. The wh-phrases are restricted to sequences of function word beginning with an interrogative, (i.e. who, what, when, where, why, and how). These wh-phrases tend to coarsely classify questions into a few types. *Tritus* uses heuristic rules and thresholds of term frequencies to learn transforms.

In contrast to previous work, we rely on a mathematical model trained on a set of questions and answers to learn how to transform the question into an effective query. Transformations are learned based on a more fine-grained question classification involving the interrogative and one or more content words.

3 Transforming Question to Query

The method is aimed at automatically learning of the best transforms that turn a given natural language question into an effective query by using the Web as corpus. To that

end, we first automatically obtain a collection of answer passages (*AP*s) as the training corpus from the Web by using a set of (Q, A) pairs. Then we identify the question pattern for each Q by using statistical and linguistic information. Here, a question pattern Q_p is defined as a question word plus one or two keywords that are related to the question word. Q_p represents the question intention and it can be treated as a preference indicative for fine-grained type of named entities. Finally, we decide the transforms Ts for each Q_p by choosing those phrases in the *AP*s that are statistically associated with Q_p and adjacent to the answer A.

Table 1. An example of converting a question (Q) with its answer (A) to a *SE* query and retrieving answer passages (*AP*)

(Q, A)	*AP*
What is the capital of Pakistan? Answer:(*Islamabad*)	Bungalow For Rent in *Islamabad*, Capital Pakistan. Beautiful Big House For …
	Islamabad is the capital of Pakistan. Current time, …
$(k_1, k_2, \ldots, k_n, A)$	
capital, Pakistan, Islamabad	…the airport which serves Pakistan's capital *Islamabad*, …

3.1 Search the Web for Relevant Answer Passages

For training purpose, a large amount of question/answer passage pairs are mined from the Web by using a set of question/answer pairs as seeds.

More formally, we attempt to retrieve a set of (Q, *AP*) pairs on the Web for training purpose, where Q stands for a natural language question, and *AP* is a passage containing at least one keyword in Q and A (the answer to Q). The seed data (Q, A) pairs can be acquired from many sources, including trivia game Websites, TREC QA Track benchmarks, and files of Frequently Asked Questions (FAQ). The output of this training-data gathering process is a large collection of (Q, *AP*) pairs. We describe the procedure in details as follows:

1. For each (Q, A) pair, the keywords k_1, k_2,…, k_n are extracted from Q by removing stopwords.
2. Submit (k_1, k_2,…, k_n, A) as a query to a search engine *SE*.
3. Download the top n summaries returned by *SE*.
4. Separate sentences in the summaries, and remove HTML tags, URL, special character references (e.g., "<").
5. Retain only those sentences which contain A and some k_i.

Consider the example of gathering answer passages from the Web for the (Q, A) pair where Q = "*What is the capital of Pakistan?*" and A = "*Islamabad*." See Table 1 for the query submitted to a search engine and potential answer passages returned.

3.2 Question Analysis

This subsection describes the presented identification of the so-called "question pattern" which is critical in categorizing a given question and transforming the question into a query.

Formally, a "question pattern" for any question is defined as following form:

question-word head-word+

where "question-word" is one of the interrogatives (Who/What/Where/When/How) and the "head-word" represents the headwords in the subsequent chunks that tend to reflect the intended answer more precisely. If the first headword is a light verb, an additional headword is needed. For instance, "*who had hit*" is a reasonable question pattern for "*Who had a number one hit in 1984 with 'Hello'?*", while "*who had*" seems to be too coarse.

In order to determine the appropriate question pattern for each question, we examined and analyzed a set of questions which are part-of-speech (POS) tagged and phrase-chunked. With the help of a set of simple heuristic rules based on POS and chunk information, fine-grained classification of questions can be carried out effectively.

Question Pattern Extraction

After analyzing recurring patterns and regularity in quizzes on the Web, we designed a simple procedure to recognize question patterns. The procedure is based on a small set of prioritized rules.

The question word which is one of the wh-words ("*who*," "*what*," "*when*," "*where*," "*how*," or "*why*") tagged as determiner or adverbial question word. According to the result of POS tagging and phrase chunking, we further decide the main verb and the voice of the question. Then, we apply the following expanded rules to extract words to form question patterns:

Rule 1: *Question word in a chunk of length more than one (see Example (1) in Table 2).*
Qp = question word + headword in the same chunk

Rule 2: *Question word followed by a light verb and Noun Phrase(NP) or Prepositional Phrase(PP) chunk (Example (2)).*
Qp = question word + light verb +headword in the following NP or PP chunk

Rule 3: *Question word followed immediately by a verb (Example (3)).*
Qp = question word + headword in the following Verb Phrase(VP) or NP chunk

Rule 4: *Question word followed by a passive VP (Example (4)).*
Qp = Question word + "to be" + headword in the passive VP chunk

Rule 5: *Question word followed by the copulate "to be" and an NP (Example (5)).*
Qp = Question word + "to be" + headword in the next NP chunk

Rule 6: *If none of the above rules are applicable, the question pattern is the question word.*

By exploiting linguistic information of POS and chunks, we can easily form the question pattern. These heuristic rules are intuitive and easy to understand. Moreover, the fact that these patterns which tend to recur imply that they are general and it is easy to gather training data accordingly. These question patterns also indicate a preference for the answer to be classified with a fine-grained type of proper nouns. In

the next section, we describe how we exploit these patterns to learn the best question-to-query transforms.

Table 2. Example questions and question patterns (of words shown in bold)

(1)	**Which** female **singer** performed the first song on Top of the Pops?
(2)	**Who** in 1961 **made** the first space **flight**?
(3)	**Who painted** "The Laughing Cavalier"?
(4)	**What is** a group of geese **called**?
(5)	**What** is the second longest **river** in the world?

3.3 Learning Best Transforms

This section describes the procedure for learning transforms *T*s which convert the question pattern Q_p into bigrams in relevant *AP*s.

Word Alignment Across *Q* and *AP*

We use word alignment techniques developed for statistical machine translation to find out the association between question patterns in *Q* and bigrams in *AP*. The reason why we use bigrams in *AP*s instead of unigrams is that bigrams tend to have more unique meaning than single words and are more effective in retrieving relevant passages.

We use Competitive Linking Algorithm [8] to align a set of (*Q*, *AP*) pairs. The method involves preprocessing steps for each (*Q*, *AP*) pair so as to filter useless information:

1. Perform part-of-speech tagging on *Q* and *AP*.
2. Replace all instances of *A* with the tag <ANS> in *AP*s to indicate the location of the answers.
3. Identify the question pattern, Q_p and keywords which are *not* a named entity. We denote the question pattern and keywords as $q_1, q_2, ..., q_n$.
4. Convert *AP* into bigrams and eliminate bigrams with low term frequency (tf) or high document frequency (df). Bigrams composed of two function words are also removed, resulting in bigrams $a_1, a_2, ..., a_m$.

We then align *q*'s and *a*'s via Competitive Linking Algorithm (CLA) procedure as follows:

Input: A collection *C* of (*Q*; *A*) pairs, where $(Q; A) = (q_1 = Q_p, q_2, q_3, ..., q_n; a_1, a_2, ..., a_m)$

Output: Best alignment counterpart *a*'s for all *q*'s in *C*

1. For each pair of (*Q*; *A*) in *C* and for all q_i and a_j in each pair of C, calculate LLR(q_i, a_j), logarithmic likelihood ratio (LLR) between q_i and a_j, which reflects their statistical association.
2. Discard (*q*, *a*) pairs with a LLR value lower than a threshold.

3. For each pair of $(Q; A)$ in C and for all q_i and a_j therein, carry out Steps 4-7:
4. Sort list of (q_i, a_j) in each pair of $(Q; A)$ by decreasing LLR value.
5. Go down the list and select a pair if it does not conflict with previous selection.
6. Stop when running out of pairs in the list.
7. Produce the list of aligned pairs for all Qs and APs.
8. Tally the counts of aligning (q, a).
9. Select top k bigrams, $t_1, t_2, ..., t_k$, for every question pattern or keyword q.

The LLR statistics is generally effective in distinguishing related terms from unrelated ones. However, if two terms occur frequently in questions, their alignment counterparts will also occur frequently, leading to erroneous alignment due to indirect association. CLA is designed to tackle the problem caused by indirect association. Therefore, if we only make use of the alignment counterpart of the question pattern, we can keep the question keywords in Q so as to reduce the errors caused by indirect association. For instance, the question "*How old was Bruce Lee when he died?*" Our goal is to learn the best transforms for the question pattern "*how old*." In other words, we want to find out what terms are associated with "*how old*" in the answer passages. However, if we consider the alignment counterparts of "*how old*" without considering those keyword like "*died*," we run the risk of getting "*died in*" or "*is dead*" rather than "*years old*" and "*age of*." If we have sufficient data for a specific question pattern like "*how long*," we will have more chances to obtain alignment counterparts that are effective terms for query expansion.

Distance Constraint and Proximity Ranks

In addition to the association strength implied with alignment counts and co-occurrence, the distance of the bigrams to the answer should also be considered. We observe that terms in the answer passages close to the answers intuitively tend to be useful in retrieving answers. Thus, we calculate the bigrams appearing in a window of three words appearing on both sides of the answers to provide additional constraints for query expansion.

Combing Alignment and Proximity Ranks

The selection of the best bigrams as the transforms for a specific question pattern is based on a combined rank of alignment count and proximity count. It takes the average of these two counts to re-rank bigrams. The average rank of a bigram b is

$$Rank_{avg}(b) = (Rank_{align}(b) + Rank_{prox}(b))/2,$$

where $Rank_{align}(b)$ is the rank of b's alignment count and $Rank_{prox}(b)$ is the rank of b's proximity count. The n top-ranking bigrams for a specific type of question will be chosen to transform the question pattern into query terms. For the question pattern "*how old*," the candidate bigrams with alignment ranks, co-occurring ranks, and average ranks are shown in Table 3.

Table 3. Average rank calculated from for the bigram counterparts of "*how old*"

Bigrams	Alignment Rank	Proximity Rank	Avg. Rank	Final Rank
age of	1	1	1	1
years old	2	2	2	2
ascend the	3	-	-	-
throne in	4	3	3.5	3
the youngest	3	-	-	-
…	…	…	…	…

3.4 Runtime Transformation of Questions

At runtime, a given question Q submitted by a user is converted into one or more keywords and a question pattern, which is subsequently expanded in to a sequence of query terms based on the transforms obtained at training.

We follow the common practice of keyword selection in formulating Q into a query:

- Function words are identified and discarded.
- Proper nouns that are capitalized or quoted are treated as a single search term with quotes.

Additionally, we expand the question patterns based on alignment and proximity considerations:

- The question pattern Q_p is identified according to the rules (in Section 3.2) and is expanded to be a disjunction (sequence of ORs) of Q_p's headword and n top-ranking bigrams (in section 3.3)
- The query will be a conjunction (sequence of ANDs) of expanded Q_p, proper names, and remaining keywords. Except for the expanded Q_p, all other proper names and keywords will be in the original order in the given question for the best results.

Table 4. An example of transformation from question into query

<table>
<tr><th colspan="3">Question</th></tr>
<tr><td colspan="3">How old was Bruce Lee when he died?</td></tr>
<tr><th>Question pattern</th><th>Proper noun</th><th>Keyword</th></tr>
<tr><td>how old</td><td rowspan="3">"Bruce Lee"</td><td rowspan="3">died</td></tr>
<tr><th>Transformation</th></tr>
<tr><td>age of, years old</td></tr>
<tr><th colspan="3">Expanded query</th></tr>
<tr><td colspan="3">Boolean query: ("old" OR "age of" OR "years old") AND "Bruce Lee" AND "died"</td></tr>
<tr><td colspan="3">Equivalent Google query: (old || "age of" || "years old") "Bruce Lee" died</td></tr>
</table>

For example, formulating a query for the question "*How old was Bruce Lee when he died?*" will result in a question pattern "*how old*." Because there is a proper noun "*Bruce Lee*" in the question and a remaining keyword "*died*," the query becomes "('*old*' OR '*age of*' OR '*years old*') AND '*Bruce Lee*' AND '*died*.'" Table 4 lists the query formulating for the example question.

4 Experiments and Evaluation

The proposed method is implemented by using the Web search engine, Google, as the underlying information retrieval system. The experimental results are also justified with assessing the effectiveness of question classification and query expansion.

We used a POS tagger and chunker to perform shallow parsing of the questions and answer passages. The tagger was developed using the Brown corpus and WordNet. The chunker is built from the shared CoNLL-2000 data provided by CoNLL-2000. The shared task CoNLL-2000 provides a set of training and test data for chunks. The chunker we used produces chunks with an average precision rate of about 94%.

4.1 Evaluation of Question Patterns

The 200 questions from TREC-8 QA Track provide an independent evaluation of how well the proposed method works for question pattern extraction works. We will also give an error analysis.

Table 5. Evaluation results of question pattern extraction

	Two "good" labels	At least one "good" label
Precision (%)	86	96

Table 6. The first five questions with question patterns and judgment

Question	Question pattern	Judgment
Who is the author of the book, "The Iron Lady: A Biography of Margaret Thatcher"?	Who-author	good
What was the monetary value of the Nobel Peace Prize in 1989?	What value	good
What does the Peugeot company manufacture?	What do manufacture	good
How much did Mercury spend on advertising in 1993?	How much	good
What is the name of the managing director of Apricot Computer?	What name	bad

Two human judges both majoring in Foreign Languages were asked to assess the results of question pattern extraction and give a label to each extracted question pattern. A pattern will be judged as "good" if it clearly expresses the answer preference of the question; otherwise, it is tagged as "bad." The precision rate of extraction for these 200 questions is shown in Table 5. The second column indicates the precision rate when both of two judges agree that an extracted question pattern is "good." In addition, the third column indicates the rate of those question patterns that are found to be "good" by either judge. The results imply that the proposed pattern extraction rules are general, since they are effective even for questions independent of the training and development data. Table 6 shows evaluation results for "two 'good' labels" of the first five questions.

We summarize the reasons behind these bad patterns:

- Incorrect part-of-speech tagging and chunking
- Imperative questions such as "*Name the first private citizen to fly in space.*"
- Question patterns that are not specific enough

For instance, the system produces "*what name*" for "*What is the name of the chronic neurological autoimmune disease which ...* ?", while the judges suggested that "*what disease*.". Indeed, some of the patterns extracted can be modified to meet the goal of being more fine-grained and indicative of a preference to a specific type of proper nouns or terminology.

4.2 Evaluation of Query Expansion

We implemented a prototype of the proposed method called *Atlas* (Automatic Transform Learning by Aligning Sentences of question and answer). To develop the system of *Atlas,* we gathered seed training data of questions and answers from a trivia game website, called QuizZone[1]. We collected the questions posted in June, 2004 on QuizZone and obtained 3,851 distinct question-answer pairs. We set aside the first 45 questions for testing and used the rest for training. For each question, we form a query with question keywords and the answer and submitted the query to Google to retrieve top 100 summaries as the answer passages. In all, we collected 95,926 answer passages.

At training time, we extracted a total of 338 distinct question patterns from 3,806 questions. We aligned these patterns and keywords with bigrams in the 95,926 answer passages, identified the locations of the answers, and obtained the bigrams appearing within a distance of 3 of the answers. At runtime, we use the top-ranking bigram to expand each question pattern. If no such bigrams are found, we use only the keyword in the question patterns. The expanded terms for question pattern are placed at the beginning of the query.

We submitted forty-five keyword queries and the same number of expanded queries generated by *Atlas* for the test questions to Google and obtained ten returned summaries for evaluation. For the evaluation, we use three indicators to measure the performance. The first indicator is the mean reciprocal rank (*MRR*) of the first relevant document (or summary) returned. If the *r*-th document (summary) returned is the one with the answer, then the reciprocal rank of the document (summary) is *1/r*.

[1] QuizZone (http://www.quiz-zone.co.uk)

The mean reciprocal rank is the average reciprocal rank of all test questions. The second indicator of effective query is the recall at R document retrieved (Recall at R). The last indicator measures the human effort (HE) in finding the answer. HE is defined as the least number of passages needed to be viewed for covering all the answers to be returned from the system.

The average length of these test questions is short. We believe the proposed question expansion scheme helps those short sentences, which tend to be less effective in retrieving answers. We evaluated the expanded queries against the same measures for summaries returned by simple keyword queries. Both batches of returned summaries for the forty-five questions were verified by two human judges.

As shown in Table 7, the MRR produced by keyword-based scheme is slightly lower than the one yielded by the presented query expansion scheme. Nevertheless, such improvement is encouraging by indicating the effectiveness of the proposed method.

Table 8 lists the comparisons in more details. It is found that our method is effective in bringing the answers to the top 1 and top 2 summaries as indicated by the high Recall of 0.8 at $R = 2$. In addition, Table 8 also shows that less user's efforts are needed by using our approach. That is, for each question, the average of summaries required to be viewed by human beings goes down from 2.7 to 2.3.

In the end, we found that those bigrams containing a content word and a function word turn out to be very effective. For instance, our method tends to favor transforms

Table 7. Evaluation results of MRR

Performances	MRR
GO (Direct keyword query for Google)	0.64
AT+GO (Atlas expanded query for Google)	0.69

Table 8. Evaluation Result of Recall at R and Human Effort

Rank	Rank count		Recall at R	
	GO	AT+GO	GO	AT+GO
1	25	26	0.56	0.58
2	6	10	0.69	0.80
3	5	3	0.80	0.87
4	0	1	0.80	0.89
5	1	1	0.82	0.91
6	2	0	0.87	0.91
7	1	0	0.89	0.91
8	2	0	0.93	0.91
9	0	1	0.93	0.93
10	0	0	0.93	0.93
No answers	3	3		
Human Effort	122	105		
# of questions	45	45		
HE per question	2.7	2.3		

such as "*who invented*" to bigrams such as "*invented by*," "*invent the*," and "*inventor of*." This contrasts to conventional wisdom of using a stoplist of mostly function words and excluding them from consideration in a query. Our experiment also shows a function word as part of a phrasal term seems to be very effective, for it indicate an implied relation with the answer.

5 Conclusion and Future Work

In this paper, we introduce a method for learning query transformations that improves the ability to retrieve passages with answers using the Web as corpus. The method involves question classification and query transformations using a learning-based approach. We also describe the experiment with over 3,000 questions indicates that satisfactory results were achieved. The experimental results show that the proposed method provides effective query expansion that potentially can lead to performance improvement for a question answering system.

A number of future directions present themselves. First, the patterns learned from answer passages acquired on the Web can be refined and clustered to derive a hierarchical classification of questions for more effective question classification. Second, different question patterns, like "*who wrote*" and "*which author*", should be treated as the same in order to cope with data sparseness and improve system performance. On the other hand, an interesting direction is the generating pattern transformations that contain the answer extraction patterns for different types of questions.

References

1. Ittycheriah, A., Franz, M., Zhu, W.-J., and Rathaparkhi, A. 2000. IBM's statistical question answering system. In Proceedings of the TREC-9 Question Answering Track, Gaithersburg, Maryland.
2. Hovy, E., Gerber, L., Hermjakob, U., Junk, M., and Lin, C.-Y. 2000. Question answering in Webclopedia. In Proceedings of the TREC-9 Question Answering Track, Gaithersburg, Maryland.
3. Moldovan D., Pasca M., Harabagiu S., & Surdeanu M. 2002. Performance Issues and error Analysis in an Open-Domain Question Answering System. In Proceedings of the 40th Annual Meeting of ACL, Philadelphia, Pennsylvania.
4. Hildebrandt, W., Katz, B., & Lin, J. 2004. Answering definition questions with multiple knowledge sources. In Proceedings of the 2004 Human Language Technology Conference and the North American Chapter of the Association for Computational.
5. Radev, D. R., Qi, H., Zheng, Z., Blair-Goldensohn, S., Fan, Z. Z. W., and Prager, J. M. 2001. Mining the web for answers to natural language questions. In Proceedings of the International Conference on Knowledge Management (CIKM-2001), Atlanta, Georgia.
6. Hermjakob, U., Echihabi, A., and Marcu, D. 2002. Natural Language Based Reformulation Resource and Web Exploitation for Question Answering. In Proceeding of TREC-2002, Gaithersburg, Maryland.
7. Agichtein, E., Lawrence, S., and Gravano, L. Learning to find answers to questions on the Web. 2003. In ACM Transactions on Internet Technology (TOIT), 4(2):129-162.
8. Melamed, I. D. 1997. A Word-to-Word Model of Translational Equivalence. In Proceedings of the 35st Annual Meeting of ACL, Madrid, Spain.
9. Yi-Chia Wang, Jian-Cheng Wu, Tyne Liang, and Jason S. Chang. 2004. Using the Web as Corpus for Un-supervised Learning in Question Answering, Proceedings of Rocling 2004, Taiwan.