

Relation-Based Document Retrieval for Biomedical IR*

Xiaohua Zhou¹, Xiaohua Hu¹, Guangren Li², Xia Lin¹, and Xiaodan Zhang¹

¹ College of Information Science & Technology, Drexel University
3141 Chestnut Street, Philadelphia, PA 19104

² Faculty of Economy, Hunan University, Changsha, China
xiaohua.zhou@drexel.edu, {thu, xlin, xzhang}@cis.drexel.edu

Abstract. In this paper, we explore the use of term relations in information retrieval for precision-focused biomedical literature search. A relation is defined as a pair of two terms which are semantically and syntactically related to each other. Unlike the traditional “bag-of-word” model for documents, our model represents a document by a set of sense-disambiguated terms and their binary relations. Since document level co-occurrence of two terms, in many cases, does not mean this document addresses their relationships, the direct use of relation may improve the precision of very specific search, e.g. *searching documents that mention genes regulated by Smad4*. For this purpose, we develop a generic ontology-based approach to extract terms and their relations, and present a betweenness centrality based approach to rank retrieved documents. A prototyped IR system supporting relation-based search is then built for Medline abstract search. We use this novel IR system to improve the retrieval result of all official runs in TREC-2004 Genomics Track. The experiment shows promising performance of relation-based IR. The average P@100 (the precision of top 100 documents) for 50 topics is significantly raised from 26.37 % (the P@100 of the best run is 42.10%) to 53.69% while the MAP (mean average precision) is kept at an above-average level of 26.59%. The experiment also shows the expressiveness of relations for the representation of information needs, especially in the area of biomedical literature full of various biological relations.

1 Introduction

Precision (the proportion of relevant documents in all retrieved documents) and recall (the proportion of retrieved relevant documents in all relevant documents in the collection) are two basic metrics to measure the performance of Information Retrieval (IR). Often, high precision is at the cost of low recall, and vice versa. Nowadays, precision-focused searching is getting more and more attention most likely due to the following two reasons. First, in a lot of domain-specific application-related search, such as searching the Medline, which collects 14 millions of biomedical abstracts published in more than 4600 journals, the biomedical professional normally know what they need and their search queries are often very specific and only like to receive

* This research work is supported in part from the NSF Career grant (NSF IIS 0448023). NSF CCF 0514679 and the research grant from PA Dept of Health.

those documents which meet their specific query, they do not expect a large number of documents. Second, the absolute number of returned relevant document is still large enough for majority of tasks even if the recall is low because of the exponentially increasing size of collections.

Term-based IR models view a document as a bag-of-term, i.e. each term is treated independently without considering the possible connections or relationships. They assign each term a weight by various methods such as TF*IDF family methods [9, 14] and language modeling methods [13] while computing the similarity between document and query. They do not explicitly address the semantics of terms either though some approaches such as latent semantic indexing [3] try to identify the latent semantic structure between terms. Basically, this line of statistical approaches is efficient and effective in IR. However, they may not be effective to approach very specific information needs that address the relationship between terms.

Term-based IR models have to use term co-occurrence to approximate term relations because there are no direct relations available in their models. However, the co-occurrence of two terms in a document, in many cases, does not mean this document really addresses their relationships, especially when the co-occurrence count is low (e.g. in abstract-based search, the co-occurrence count is often low). Thus, the precision would be compromised. We conducted a simple experiment that tried to retrieve documents addressing the interaction of *obesity* and *hypertension* from PubMed¹ by specifying the co-occurrence of term *hypertension* and *obesity* in abstract or title. We then took the top 100 abstracts for human relevance judgment. Unfortunately, as expected, only 33 of them were relevant.

*obesity [TIAB] AND hypertension [TIAB] AND hasabstract [text]
AND ("1900"[PDAT] : "2005/03/08"[PDAT])*

Fig. 1. The query used to retrieve documents addressing the interaction of obesity and hypertension from PubMed. A ranked hit list of 6687 documents is returned.

Based on this finding, we develop a precision-focused IR model for domain-specific search, which basically treats a document a set of sense-disambiguated terms and their binary relations. A relation is defined as a pair of two terms which are semantically and syntactically related to each other. Since a relation in our model is explicitly asserted, the direct use of relation in IR may improve the precision of domain-specific search though the recall may be slightly lowered.

Retrieval of biomedical literature often involves various specific biological relations. Take the example of TREC 2004 Genomics Track [7] the goal of which is to study retrieval tasks in genomic domain. All 50 ad hoc retrieval topics² are compiled from real information needs of scientists in biomedical domain and most of them are about very specific relationships among gene (protein), mutations, genetic functions, diseases and so on (see some examples in Section 2.2). For this reason, relation-based IR is an appropriate approach to biomedical literature search.

¹ <http://www.ncbi.nlm.nih.gov/entrez/query.fcgi>

² <http://trec.nist.gov/data/genomics/04.adhoc.topics.txt>

The extraction of binary relations from text is a challenging task. We think this is one of the reasons that there is no relation-based approach to IR reported so far. Term extraction is the first step of relation extraction. The methods for term extraction fall into two categories, with dictionary [12, 20] or without dictionary [11, 17, and 18]. The later is characterized by its high extracting speed and no reliance on dictionary. However, it does not identify meaning (sense) of a term which is important to our IR model. For this reason, we apply a dictionary-based approach [20] to the extraction of term. Majority of the literature use patterns learned by either supervised [11] or unsupervised approaches [8, 12] to identify binary relations. But almost all these approaches are only tested on extraction of protein-protein interactions. Besides, their extracting recall is too low for IR use. We finally develop a generic ontology-based approach to extract terms and their binary relations.

Ranking is an important component to IR systems. Most existing ranking methods are directly or indirectly based on term frequency. However, frequency does not well capture the structure of terms and relations in a document. For this reason, we try to find a better metric that can fully use the information of binary relations between terms. Considering a document in our IR model can be easily formalized as a graph, $G = (V, E)$, where V denotes all terms and E denotes all binary relations, we borrow approaches and metrics from social network research [1, 6] and develop a betweenness centrality based ranking approach.

Based on the above ranking and extracting approaches, we build a prototyped IR system supporting relation-based search for Medline abstracts. We use this novel IR system to improve the retrieval result of all official runs in TREC-04 Genomics Track. The experiment shows promising performance of relation-based IR. The average $P@100$ (the precision of top 100 documents) for 50 topics is significantly raised from 26.37 % (the $P@100$ of the best run is 42.10%) to 53.69% while the MAP (mean average precision) is kept at an above-average level of 26.59%. The $P@10$ is also improved from 42.69% to 61.87%. The experiment also shows the expressiveness of relations for the representation of information needs, especially in the area of biomedical literature which are full of various biological relations.

The rest of the paper is organized as follows: Section 2 describes the representation of documents. Section 3 presents a generic approach to extraction of terms and relations. Section 4 shows a ranking approach for relation-based IR model. Section 5 presents the experiment design and result. A short conclusion finishes the paper.

2 Representation of Document and Query

Traditional IR models a document as a bag-of-word (a), i.e. a document consists of a set of words which are treated as independent of each other. Because a term (it is also called phrase in other work) is often more meaningful than a word, bag-of-term model (b) is naturally extended from the bag-of-word model. For example, *high blood pressure* is treated as one token instead of three tokens in bag-of-term model. A word or a term may have different meanings in different context. Thus, a bag-of-sense (c) is further evolved for information retrieval.

The above three models (a, b, and c) may produce slightly different performance for IR. But neither of them addresses the relation among tokens. Actually, a document

is often full of various explicit and implicit relations. For example, biomedical literatures contain a large number of biological interactions among gene, protein, mutation, disease, drug, etc. Intuitively, incorporation of such knowledge (represented by relations) will help improve the performance of an IR system. For this purpose, we propose a relation-based IR model below.

Terms (CUI, String, Semantic Type, Frequency)	
T1 (C0003818, arsenic, Hazardous or Poisonous Substance, 9)	
T2 (C0870082, hyperkeratosis, Disease or Syndrome, 4)	
T3 (C1333356, XPD, Gene, 6)	
T4 (C0007114, skin cancer, Neoplastic Process, 1)	
T5 (C0012899, DNA repair, Genetic Function, 3)	
T6 (C0241105, hyperkeratotic skin lesion, Finding, 2)	
T7 (C0936225, inorganic arsenic, Inorganic Chemical, 1)	
.....	
Relations (First Term, Second Term, Frequency, Type)	
R1 (T1, T3, 3, E)	R2 (T2, T4, 1, E)
R3 (T2, T5, 2, E)	R4 (T2, T3, 2, E)
R5 (T4, T5, 1, E)	R6 (T3, T4, 1, E)
.....	

Fig. 2. A real example of document representation. The document (PMID: 12749816) can be found through PubMed. CUI is the sense ID of a concept in UMLS³. *E* in relation representation stands for entity-entity relation.

2.1 Document Representation

In relation-based IR model, we represent a document by a set of sense disambiguated terms and their binary relations as shown in Figure 2. We record the sense rather than the string as the unique identity of a term based on the following two considerations. First, term sense can relieve the synonym problem in IR. Because all synonyms share one sense ID, we can simply use one sense ID to find all documents containing its synonyms without query expansion. Second, it can solve polysemous problem in IR because a word (even a phrase) may have different meanings across documents and queries while the sense ID never causes ambiguity [15, 19]. However, strings still provide useful information for IR. For example, in the experiment of TREC 2004 Genomics Track (see Section 2.2), we use string to decide if a term (protein) belongs to certain protein family. Thus, we keep the string of a term in term indices. Also, we record the semantic type of a term, the category a term belongs to. The semantic type is useful to express information needs (see Section 2.2).

A relation is defined as a pair of two terms which are semantically and syntactically related to each other. We identify all such term pairs in a document and record their frequency. The relations fall into two types: entity-entity relation and entity-attribute relation. The entity-entity relation addresses the interaction of two entities,

³ <http://www.nlm.nih.gov/research/umls/>

for example, the protein-protein interaction and the relation between genes and diseases. For the simplicity, the entity-entity relation in our model is undirected. The other type of relation is entity-attribute. It is about from what point of view the entity is described. For example, in the entity-attribute relation, regulation of TGF β gene, TGF β gene is the entity and regulation is the attribute of TGF β gene. Obviously, the entity-attribute relation is directed.

2.2 Query Representation

The query representation is subject to the mechanism of document representation. Under traditional term-based IR model, we often use term vector or term-based Boolean expression to represent information needs. In this section, we will first briefly introduce the syntax of relation-based Boolean expression and then demonstrate the effectiveness of this query representation mechanism by the examples from TREC 2004 Genomics Track.

Three types of predicates, denoted by term (T), entity-entity relation (R), and entity-attribute relation (M), are available to build Boolean expression. A term can be specified by any combination of its string (STR), sense ID (CUI), and semantic type (TUI). All predicates can be combined by AND or OR operator. Here, we use the ad hoc topics in TREC 2004 Genomics Track⁴ to illustrate the usage of relation-based Boolean expression to represent user information needs.

Topic #1: Ferroportin-1 in humans

Query: T (CUI=C0915115)

Notes: C0915115 is the sense ID of *Ferroportin-1* in the dictionary of UMLS (Unified Medical Language System). All term senses in this paper is based on UMLS.

Topic #2: Generating transgenic mice

Query: M (CUI₁=C0025936 AND STR₂=generation)

Notes: C0025936 is the sense ID of transgenic *mice*

Topic #12: Genes regulated by Smad4

Query: R (CUI₁=C0694891 and TUI₂=T028)

Notes: C0694891 is the sense ID of *Smad4* and T028 stands for the semantic type if *Gene*. Because entity-entity relation is undirected, the query should contain the symmetric predicate R (CUI₂=C0694891 and TUI₁=T028). However, for the simplicity, we let the IR system automatically generate the symmetric predicate R.

Topic #14: Expression or Regulation of TGF β in HNSCC cancers

Query: R (CUI₁=C1515406 and CUI₂=C1168401)

Notes: C1515406 is the sense ID of *TGF β* and C1168401 is the sense ID of *HNSCC*

Topic #30: Regulatory targets of the Nkx gene family members

Query: R (STR₁ like nkx% and TUI₁=T028 and TUI₂=T028)

Notes: we assume a term with its string beginning with nkx and with semantic type of gene is the member of *Nkx gene family*.

⁴ <http://trec.nist.gov/data/genomics/04.adhoc.topics.txt>

We can see that relation-based Boolean expression is neat and powerful to express user information needs from above examples. In topic #1, we simply use one T predicate though Ferroportin-1 has lots of synonyms. In topic #12 and #30, we use one R predicate in conjunction with semantic types to express a question-answering type information need that is very difficult to be represented by term vector or term-based Boolean expression.

3 Extraction of Terms and Relations

In this section, we propose a generic ontology-based approach to extraction of terms and relations. As shown in Figure 3, we first extract terms using domain ontology in conjunction with part of speech patterns [20]; then use surrounding words to narrow down the sense; finally employ several heuristic approaches to generate relations.

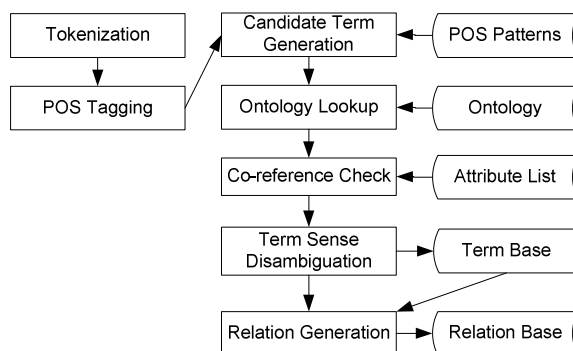


Fig. 3. The architecture of the term and relation extraction system

3.1 Extraction of Terms

There are volumes of literature on the topic of term extraction from biomedical literatures. Most of them use either hand-created rules or machine-learned rules to extract terms from text. However, neither of them extracts meaning (sense) of the term. For instance, the IE system may tell you that Ferroportin-1 is a protein but not tell you what protein it is. Because we record sense ID rather than string as the ID of a term, we use a generic ontology-based approach [20] that identify not only the category of the term, but also its possible senses. This approach begins with part of speech (POS) tagging, then generates candidate terms using POS patterns, and finally determines if it is a term by looking up the ontology.

In this particular project, we take UMLS as the domain ontology. UMLS is built from the electronic versions of many different thesauri, classifications, code sets, and lists of controlled terms in the area of biomedicine and health. The Metathesaurus of UMLS is organized by concept or meaning of terms and provides their various names (synonyms), and the relationships among them. By checking with the synonym table, we can easily determine if the candidate (generated by POS patterns listed in Table 1) is a term and retrieve possible senses if yes.

Table 1. Part of Speech Patterns and Examples. NN, NUM, and JJ denote noun, number, and adjective, respectively. All article, preposition, and conjunction words will be removed from the original text during pattern matching.

Part of Speech Pattern	Examples
NN NN NN	Cancer of Head and Neck
NN NUM NN	DO 1 Antibody
JJ NN NN	High Blood Pressure
NN NN	DNA Repair
NN NUM	Ferroportin 1
JJ NN	Sleeping Beauty
NN	FancD2

A term sometimes appears in the form of a pronoun such as *it* or its abbreviation. It is then necessary to figure out what the pronoun or abbreviation refers to in context. We develop a simple heuristic approach to handle abbreviations and implement a light method provided by Dimitrov and his colleagues [4].

3.2 Term Sense Disambiguation

A term may have multiple meanings defined in the dictionary. For example, *Ferroportin-1* has two senses defined in UMLS (*C0915115: metal transporting protein 1*; *C1452618: Slc40a1 protein, mouse*). Inspired by a finding that the ambiguity of many terms in text is caused by use of short name, abbreviation, or partial name, we take an unsupervised sense disambiguation approach adapted from Lesk's word sense disambiguation (WSD) approach that basically tags sense by maximizing the number of common words between the definition of candidate senses and the surrounding words of the target [10].

Different from Lesk's approach, our approach first use surrounding words (3 words in the left side of the target and 3 word in the right side of the target) to narrow down candidate senses. If there is still more than one sense left, we then score each candidate sense. In Lesk's approach, any word in any sense has same weight. Obviously it is not a good assumption for term sense disambiguation. Instead, we borrow the idea from term weighting community and use TF*IDF to score the importance of a word in a sense [9, 14]. Then the final formula to tag the sense is:

$$S = \arg \max_j \sum_i IDF_i \times TF_{ij} = \arg \max_j \sum_i \log \frac{N}{n_i} \times \frac{F_{ij}}{F_j}$$

Where:

N is the number of senses in dictionary

n_i is the number of senses containing $Word_i$

F_{ij} is the occurrence of $Word_i$ in various names of $Sense_j$

F_j is the total occurrence of words in various names of $Sense_j$

3.3 Extraction of Relations

A relation is defined as a pair of two terms which are semantically and syntactically related to each other. If there is a pre-defined relation between the semantic types of two terms in domain ontology, these two terms are simply viewed as semantically related. However, the judgment of syntactic relation between two terms is difficult. We provide two different methods of syntax judgment for entity-attribute relation and entity-entity relation, respectively.

3.3.1 Entity-Attribute Relation

If two terms within one sentence match the following pattern where *term1* is in the list of candidate attributes and the preposition is either *of* or *for*, we take *term1* as the attribute of *term2*.

Rule for entity-attribute relation: *term1* preposition *term2*

Example: *Obesity is an independent risk factor (term1) for periodontal disease (term2).*

The list of candidate attribute is compiled in a semi-automatic manner. Applying the above pattern to a sample of the collection, we obtain a list of *term1* (candidate attributes). We take the *term1* with its frequency above threshold as candidates and then have one domain expert judge its qualification for being an attribute.

3.3.2 Entity-Entity Relation

Extraction of biological interactions (relations) is a hot topic in the area of information extraction. The essence of this line of work is to generalize the syntactic form of certain relation in supervised or unsupervised manner. However, there are two major problems while applying these methods to extract biological relations for IR use. First, the indexing component of our IR model is interested in many biological relationships. But most of these reported extracting methods are tested on mere protein-protein interaction. Second, the recall of these extracting methods seems to low for IR use. For example, the IE system reported by [12] only extracts 53 relationships with 43 correct from 1,000 Medline abstracts containing the keyword “protein interaction”. Instead, we employ a simple but effective heuristic approach that uses clause level co-occurrence to determine the syntactic relation of a term pair and it is able to identify various relationships with high recall and good precision for IR use.

Term co-occurrence is frequently used to determine if two terms are connected in graph-based data mining. Literature [16] takes any pair of two words in same sentence as a relation. However, as reported by [5], sentences in Medline abstracts are often very long and complex. Thus, if we follow the strategy of [16], many noisy relations may be introduced. Instead, we take clause as the boundary of a relation because terms within a clause are more cohesive than within a sentence in general. In example 1, there are three entity terms underlined and one relation (*obesity* and *periodontal disease*). The term *epidemiological study* has no relation with any of the other two terms because it is in a separate clause.

Rule for entity-entity relation: *If two terms are co-occurred within a clause, but are not coordinating components, and their semantic types are related to each other in domain ontology, this term pair is identified as an entity-entity relation.*

Example 1: *A recent epidemiological study revealed that obesity is an independent risk factor for periodontal disease.*

Example 2: *Diabetes is associated with many metabolic disorders including insulin resistance, dyslipidemia, hypertension and atherosclerosis.*

Also, Ding et al. [5] identify that coordinating is frequently occurred phenomenon in sentences and interactions (relations) between coordinating components is rare in Medline abstract. Thus, in example 2, *diabetes* has relations with remaining four terms respectively. But *insulin resistance*, *dyslipidemia*, *hypertension*, and *atherosclerosis* don't have relations with each other because they are coordinating components.

In short, we consider a term pair an entity-entity relation if these two terms are co-occurred within a clause, but are not coordinating components, and their semantic types are related to each other in domain ontology.

4 Ranking Approach

Matching the relation-based Boolean query and the relation-based representation of document, we can get a hit list for a specific query. But we still do not know the relative confidence of each document in the hit list being relevant to the query. In this section, we would answer this question, i.e. the ranking of matched documents.

A large number of term weighting schemas have been developed within TF*IDF family. The basic idea of the TF*IDF method is to synthesize the local importance of a term in a document and the global importance of a term in the collection. In general, they use inversed document frequency (IDF) to measure the global importance and use term frequency to indicate the local importance. Following this idea, we present the following framework to rank matched document:

$$R_q(d) = \sum_{p \in q} \omega_q(p) G(p) L_d(p)$$

Where $R_q(d)$ is the relevance of document d to query q , p is the predicate (term, entity-entity relation, or entity-attribute relation) that forms query, $\omega_q(p)$ is the weight of p in the query, $G(p)$ is the global importance of p in the collection, $L_d(p)$ is the local importance of p in document d .

For $\omega_q(p)$, we empirically set 0.4 for term (T), 1.0 for entity-entity relation (R), and 0.7 for entity-attribute relation (M). We still use IDF to measure the global importance of p . However, we take a metric other than *frequency* to measure the local importance. The frequency of terms or relations, of course, could be a metric of local importance because intuitively frequency is in proportion to the importance. However, frequency does not capture the structure of terms and relations in the document. That is the reason we try to find a better metric.

Since a document in our IR model is represented by a set of terms and their binary relations, it is very easy to formalize it as a graph (network), $G = (V, E)$, where V

denotes all terms and E denotes all binary relations in the document. Then we can borrow approaches and metrics in the area of social networks to measure the importance of terms (equivalent to an actor in social network) and relations (equivalent to a link in social network).

Betweenness Centrality is a frequently used metric in social network to compute the importance of an actor (a node in the network) [1, 6] and it could be extended to indicate the importance of a link (an edge in the network). The basic notion of betweenness centrality is that a vertex that can reach others on relatively short paths is relatively important. The formal definition is presented below:

In graph $G = (V, E)$, let $\sigma_{st} = \sigma_{ts}$ denotes the number of shortest paths from $s \in V$ to $t \in V$, $\sigma_{st}(v)$ denotes the number of shortest paths from s to t where some $v \in V$ lies on and $\sigma_{st}(e)$ denotes the number of shortest paths from s to t where some $e \in E$ lies on. Then the importance of a node v is defined as:

$$C_B(v) = \sum_{s \neq t \neq v} \frac{\sigma_{st}(v)}{\sigma_{st}} \quad (\text{Freeman, 1977; Anthonisse, 1971})$$

Similarly, the importance of a link e can be defined as:

$$C_B(e) = \sum_{s \neq t \neq v} \frac{\sigma_{st}(e)}{\sigma_{st}}$$

Normalization and edge weighting are two important issues while using betweenness centrality metrics. To control for the size of the network, both $C_B(v)$ and $C_B(e)$ will be normalized to lie between zero and one. Many social network researchers would like to normalize the betweenness centrality score by dividing the score by $(n-1)(n-2)/2$ where n is the number of nodes in the network. However, considering our purpose is to indicate the local importance of a term or relation

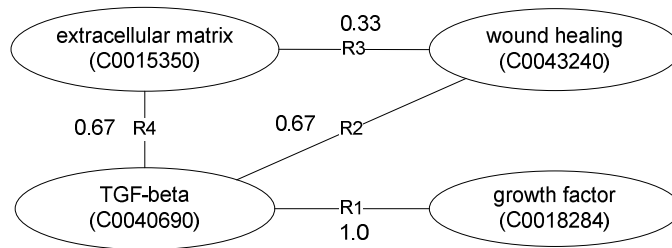


Fig. 4. A real example to calculate the local importance of relations. The document (PMID: 7929624) can be found through PubMed. The original betweenness centrality scores for relation R1, R2, R3 and R4 are 3, 2, 1 and 2, respectively. After normalization, their importance scores are 1.0, 0.67, 0.33 and 0.67.

in a document, we normalize term score and relation score by dividing their maximum value in the document, respectively. That is, the score of the most important term or relation in a document is always one.

The calculation of betweenness centrality score is related to the shortest path. Thus the weight of each edge will affect the final score. Realizing that the frequency of a relation is also an indicator to the strength or importance of the relation and related terms, we set the weight of an edge (a relation) as the inverse of the relation's occurring frequency in a document.

In short, we present a comprehensive method to rank matched documents. We consider not only the global importance of a term or a relation in the whole document collection, but also their local importance (relative importance in a document). When computing the local importance, we take into account both the structure and frequency information. The calculation of betweenness centrality score in our experiment is done by a software package JUNG⁵ that implemented a fast algorithm for betweenness centrality developed by Brandes [2].

5 Experiment

In this section, we discuss the search engine and document collection used for experiment and the experiment design. Then we analyze the experiment result and compare the performance of proposed relation-based IR model with other work.

5.1 Search Engine and Collection for Experiment

To our best knowledge, there is no search engine supporting relation-based search. For this reason, we developed a prototyped IR system supporting relation-based Boolean search. We implemented conceptual document representation in Figure 2 with a DB2 database. When a query represented by relation-based Boolean expression (see Section 2.2) is submitted, the system automatically converts the Boolean expression to ANSI SQL statement and submits the SQL statement to the DB2 system.

We use the collection of TREC 2004 Genomics Track in our experiment. The document collection is a 10-year subset (1994-2003, 4.6 million documents) of the MEDLINE bibliographic database of the biomedical literature that can be searched by PubMed. Relevance judgments were done using the conventional "pooling method" whereby a fixed number of top-ranking documents from each official run were pooled and provided to an individual for relevance judgment. The pools were built from the top-precedence run from each of the 27 groups. They took the top 75 documents for each topic and eliminated the duplicates to create a single pool for each topic. The average pool size (average number of documents judged per topic) was 976, with a range of 476-1450. Based on the human relevance judgment, the performance of each official run could be evaluated (All facts and evaluation result of TREC-04 Genomics Track in Section 5 are from [7]).

Since our goal is to see whether our relation-based IR methods can further improve TREC 2004 participants' retrieval results, we build our search engine on top of search

⁵ <http://jung.sourceforge.net/>

engines participated in TREC 2004. For this, we take the documents in pools for each topic and eliminate repeated documents across topics to create a single pool for our experiment use. The indexing and searching of our prototyped IR system is based on this mini-pool containing 42,255 documents.

5.2 Experiment Design

Our goal is to build a precision-focused IR system. The major research question of this paper is *if relation-based IR outperforms term-based IR in terms of precision*. For this reason, we compare the P@100 (the precision of top 100 documents) and P@10 of our run with that of all 47 official runs participated in TREC 2004 Genomics Track. Though the overall precision (the precision of all retrieved documents) is a good proxy for precision, we do not compare this metric because TREC did not report overall precision. For convenience, we use RIR (relation-based IR) to denote our run and TREC to denote all runs in TREC 2004 Genomics Track later.

The argument of this paper that relation-based IR outperforms term-based IR in terms of precision is actually based on the assumption that explicit assertion of term relation is more useful than document level term co-occurrence to judge whether a document addresses certain relationships. To test the truth of this assumption, we study if $R(t_1, t_2)$ provides higher precision than $T(t_1)$ and $T(t_2)$ in our experiment.

We are also interested in the recall of relation-based IR though it is not our focus. On one hand, the use of relation will lower the recall because the number of documents returned by $R(t_1, t_2)$ is always equal or less than by $T(t_1)$ and $T(t_2)$. On the other hand, the use of sense instead of string well solves the synonym problem; thus it may increase the recall. So we will study the effect of use of sense and relation on the recall of IR.

5.3 Analysis of Experiment Result

Our run retrieves 125 documents on average and achieves 53.29% overall precision, 44.31% overall recall and 26.59% MAP (Mean Average Precision). MAP is a comprehensive indicator of IR performance that captures both precision and recall. As expected, the MAP of our run is at above average level. Actually it would be ranked as 15th among all 47 official runs in TREC. Our relation-based IR system can not achieve excellent MAP currently because, (1) the system is precision-focused, (2) no query expansion method is used, and (3) it uses Boolean search rather than similarity-based search. We will take (2) and (3) for future work.

The experiment shows that relation-based IR model is effective to improve the precision. We first compare the P@100 of our run with TREC runs on 50 individual topics. Except for topic 16, the P@100 of ours outperforms the average P@100 of TREC on all other 49 topics as shown in Fig. 5. The paired-sample T test ($M=27.33\%$, $t=7.413$, $df=49$, $p=0.000$) shows the significant improvement of precision. Then we compare the P@100 of our run with P@100 of all official runs in TREC. As shown in Table 2, the P@100 of our run (53.69%) is significantly higher than that of the top 3 runs and the mean of all official runs (26.37%). The comparison of P@10 also supports the above conclusion. The average P@10 of TREC runs is significantly improved raised from 42.69% to 61.67%. It is worth noting that we can

not say that the precision of our IR system is better than that of other IR systems because our search is based on the returns of all other IR systems. But the experiment result really tells us that the relation-based model is very promising for IR because it significantly improves the result of other IR systems.

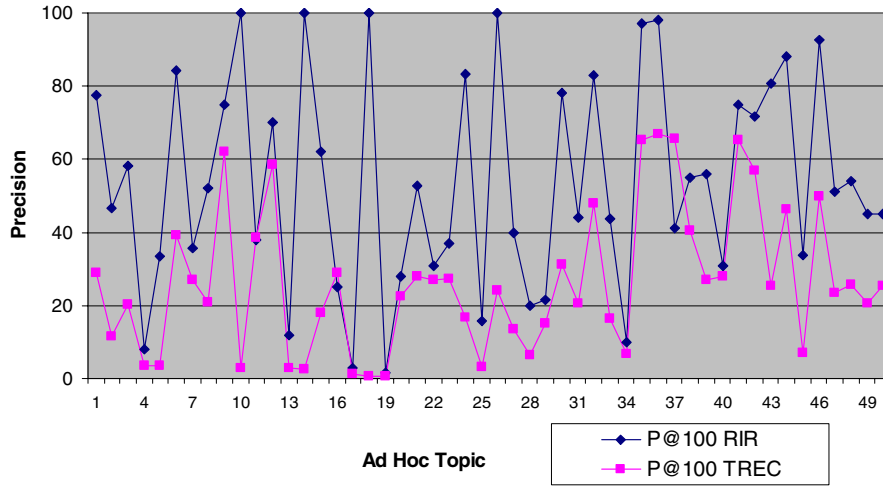


Fig. 5. The comparison of the P@100 of our run with the average P@100 of all official runs in TREC 2004 Genomic Track on 50 ad hoc retrieval topics

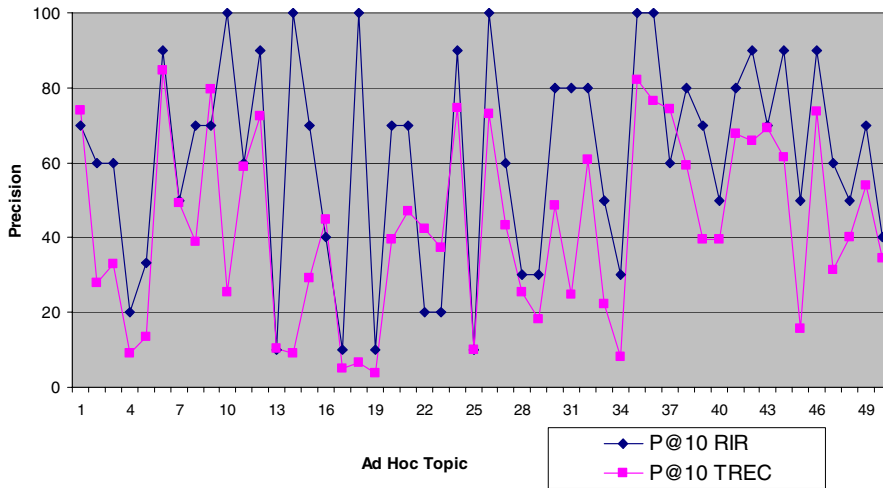


Fig. 6. The comparison of the P@10 of our run with the average P@10 of all official runs in TREC 2004 Genomic Track on 50 ad hoc retrieval topics

Table 2. The comparison of the precision of our run with official runs participated in TREC 2004 Genomics Track. Runs in TREC are sorted by Mean Average Precision (MAP) [7].

Run	MAP	P@10	P@100
Relation IR (Our Run)	26.59	61.67	53.69
pllsgen4a2 (the best)	40.75	60.04	41.96
uwntDg04tn (the second)	38.67	62.40	42.10
pllsgen4a1 (the third)	36.89	57.00	39.36
PDTNsmp4 (median)	20.74	40.56	23.18
edinauto5 (the worst)	0.12	0.36	1.3
Mean@TREC04 (47 runs)	21.72	42.69	26.37

The rationale of relation-based IR is based on the assumption that binary relation between terms provides higher precision than document-level term co-occurrence when retrieving documents addressing certain relationships. To test the truth of this assumption, we design a simple experiment to verify. For seven queries that use a single relation (R predicate) like $R(CUI_1=A \text{ and } CUI_2=B)$, we change the query to the co-occurrence of two terms, i.e. $T(CUI=A)$ and $T(CUI=B)$, and search again. The experiment result is shown in Table 3. A paired-sample T test ($M=11.68\%$, $t=4.771$, $df=6$, $p=0.003$) shows that the precision of relation-based query is significantly higher than that of term co-occurrence based query. This is the foundation of the argument of the whole paper that relation-based IR model contributes higher precision to domain-specific research than term-based IR models.

Table 3. The comparison of the use of relation and term co-occurrence in IR

Topic	$R(t_1, t_2)$		$T(t_1) \text{ and } T(t_2)$		P@100 TREC04 (%)
	P (%)	R (%)	P (%)	R (%)	
7	35.71	8.70	24.62	27.83	27.04
8	52.00	8.07	41.05	24.22	20.94
13	12.00	12.50	8.77	20.83	2.74
14	100.00	23.81	80.00	23.81	2.70
15	61.90	14.44	48.08	27.78	18.00
21	71.43	18.75	52.83	35.00	27.96
22	29.22	46.19	25.14	65.71	27.09

Table 4. The comparison of sense-based search and string-based search

Topic	String B	T(CUI=A)		T(STR like %B%)		T(STR=B)	
		P (%)	R (%)	P (%)	R (%)	P (%)	R (%)
1	Ferroportin	77.59	56.96	84.62	41.77	88.46	29.11
6	FancD2	84.09	39.36	84.09	39.36	85.29	30.85
9	mutY	73.38	98.26	81.75	97.39	81.48	95.65
35	WD40	97.16	63.10	99.28	50.55	98.28	21.03
36	RAB3A	98.10	81.50	98.10	81.50	98.53	79.13
43	Sleeping Beauty	80.56	14.87	77.42	12.31	77.42	12.31
46	RSK2	92.59	12.69	82.76	12.18	89.47	8.63

Table 5. The comparison of our run with runs in TREC on MAP, P@10, and P@100

Topic	Pool	DP	Hits	Rel.	MAP (%)		P@10		P@100	
					RIR	TREC	RIR	TREC	RIR	TREC
1	879	79	58	45	56.96	30.73	70.00	73.83	77.59	28.91
2	1264	101	30	14	11.88	5.79	60.00	27.87	46.67	11.66
3	1189	181	36	21	11.60	9.50	60.00	32.98	58.33	20.40
4	1170	30	167	10	56.67	2.98	20.00	8.94	8.00	3.60
5	1171	24	3	1	8.33	5.64	33.33	13.40	33.33	3.49
6	787	94	44	37	39.36	39.93	90.00	84.68	84.09	39.38
7	730	115	28	10	8.70	20.06	50.00	49.36	35.71	27.04
8	938	161	25	13	8.07	9.75	70.00	38.72	52.00	20.94
9	593	115	154	113	98.26	61.14	70.00	79.57	75.00	61.96
10	1126	4	3	3	75.00	58.11	100.0	25.32	100.00	2.77
11	742	111	215	85	76.58	32.69	60.00	58.94	38.00	38.43
12	810	256	255	174	67.58	42.25	90.00	72.32	70.00	58.66
13	1118	24	25	3	12.50	2.88	10.00	10.21	12.00	2.74
14	948	21	5	5	23.81	4.79	100.0	8.94	100.00	2.70
15	1111	90	21	13	14.44	13.88	70.00	29.15	61.90	18.00
16	1078	147	24	6	4.08	19.26	40.00	44.89	25.00	28.83
17	1150	3	66	2	66.67	8.85	10.00	5.11	3.03	1.15
18	1392	1	1	1	100.00	62.54	100.0	6.60	100.00	0.72
19	1135	1	63	1	100.00	15.94	10.00	3.62	1.59	0.62
20	814	116	154	33	26.72	14.66	70.00	39.57	28.00	22.38
21	676	80	53	28	18.75	26.71	70.00	47.02	52.83	27.96
22	1085	210	332	97	44.76	13.54	20.00	42.34	31.00	27.09
23	915	158	84	31	18.35	18.35	20.00	37.45	36.90	27.47
24	952	26	24	20	76.92	59.70	90.00	74.68	83.33	16.85
25	1142	32	38	6	18.75	3.31	10.00	10.00	15.79	3.30
26	792	47	9	9	19.15	44.01	100.0	72.98	100.00	24.11
27	755	29	60	24	82.76	26.40	60.00	43.19	40.00	13.55
28	836	13	60	12	92.31	20.31	30.00	25.32	20.00	6.43
29	756	43	42	9	20.93	13.52	30.00	18.09	21.43	15.15
30	1082	165	140	104	63.03	21.16	80.00	48.72	78.00	31.13
31	877	138	84	37	26.81	9.56	80.00	24.89	44.05	20.72
32	1107	496	386	323	65.12	18.04	80.00	60.85	83.00	47.87
33	812	64	39	17	26.56	13.96	50.00	22.34	43.59	16.47
34	778	31	159	19	61.29	6.44	30.00	8.30	10.00	6.68
35	717	271	176	171	63.10	34.81	100.0	82.13	97.00	65.28
36	676	254	211	207	81.50	48.87	100.0	76.38	98.00	67.00
37	476	149	250	110	73.83	53.45	60.00	74.26	41.00	65.64
38	1165	423	177	89	21.04	14.00	80.00	59.15	55.00	40.43
39	1350	317	204	107	33.75	9.84	70.00	39.36	56.00	26.89
40	1168	277	121	40	14.44	10.80	50.00	39.36	31.00	27.96
41	880	582	637	472	80.93	33.56	80.00	67.66	75.00	65.21
42	1005	697	95	68	9.76	15.87	90.00	65.96	71.58	57.02
43	739	195	36	29	14.87	11.85	70.00	69.15	80.56	25.53
44	1224	649	477	402	62.25	13.23	90.00	61.49	88.00	46.32
45	1139	156	95	32	20.51	2.86	50.00	15.74	33.68	7.11
46	742	197	27	25	12.69	26.30	90.00	73.62	92.59	49.81
47	1450	365	318	176	49.59	6.73	60.00	31.49	51.00	23.55
48	1121	155	202	104	67.10	17.12	50.00	40.21	54.00	25.57
49	1100	73	128	61	83.56	22.79	70.00	54.04	45.00	20.49
50	1091	302	174	72	23.84	7.31	40.00	34.47	45.00	25.34
Mean	975	165	125	70	26.59	21.72	61.67	42.69	53.69	26.37

Sense-based search can raise the recall of IR especially when a term has lots of synonyms because all synonyms share one sense ID. To test this hypothesis, we design the following small experiment. For seven single-term (T predicate) queries listed in Table 4, we compare the recall of sense-based search with string-based search. As expected, for the recall of topic 1 and 35, sense-based search is significantly higher than that of string-based search because both of them have many synonyms.

In this section, we successfully tested our major hypothesis that our relation-based IR model would outperform term-based IR models in terms of precision for domain-specific search. Furthermore, we tested the truth of the assumption of the major hypothesis, i.e. binary relation between terms would provide higher precision than term co-occurrence when retrieving documents addressing certain relationships. Last, we found that sense-based search would contribute higher recall to IR than string-based search especially when the searching term has many synonyms.

6 Conclusions and Future Work

In this paper, we proposed a novel relation-based information retrieval model for biomedical literature search. Unlike traditional term-based IR models that use term to index and search documents, our relation model uses sense disambiguated terms and their binary relations to index and search documents. We further develop a betweenness centrality based ranking approach that captures both frequency and structure of terms and relations. Because relations provide much contextual information and domain knowledge for IR, the use of relation may improve the precision of domain-specific IR. The experiment on a subset of the document collection of TREC 2004 Genomics Track successfully tested this hypothesis. Besides, we can draw another three conclusions from the experiment:

- An explicitly asserted relation in text is a stronger indicator of a document that addresses certain relationships between terms than the document level term co-occurrence.
- Sense-based search will bring higher recall than string-based search especially when a searching term has many synonyms.
- Relation-based Boolean expression is powerful and effective to express domain-specific information needs.

For future work, we will continue to refine the ranking approach. Though our focus is the precision, we still pay our attention to the comprehensive performance of the IR system. For this reason, we will try to extend the Boolean search to similarity-based search and find appropriate query expansion approach for relation-based IR model. Both of them will improve the recall of the IR system. We will also take effort on term and relation extraction that would further improve the performance of relation-based search.

References

1. Anthonisse, J. M., "The rush in a directed graph", Technical Report BN 9/71, Stichting Mathematisch Centrum, Amsterdam, 1971.
2. Brandes, U., "A faster algorithm for Betweenness centrality", *Journal of Mathematical Sociology*, 2001, 25(2), 163-177.

3. Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R., "Indexing by latent semantic analysis", *Journal of the American Society for Information Science*, 1990, 41(6), pp. 391-407.
4. Dimitrov, M., Bontcheva, K., Cunningham, H., and Maynard, D., "A Light-weight Approach to Coreference Resolution for Named Entities in Text", *Proceedings of the Fourth Discourse Anaphora and Anaphor Resolution Colloquium (DAARC)*, Lisbon, 2002.
5. Ding, J., Berleant, D., Xu, J., and Fulmer, A.W., "Extracting Biochemical Interactions from MEDLINE Using a Link Grammar Parser", *In the 15th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'03)*, 2003.
6. Freeman, L. C., "A set of measures of centrality based on Betweenness", *Sociometry*, 1977, 40:35-41.
7. Hersh W, et al. "TREC 2004 Genomics Track Overview", The thirteenth Text Retrieval Conference, 2004.
8. Hu, X., Yoo, I., Song, I.Y., Song, M., Han, J., and Lechner, M., "Extracting and Mining Protein-Protein Interaction Network from Biomedical Literature", *Proceedings of the 2004 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, 2004.
9. Jones, K.S., "Exhaustivity and specificity", *Journal of Documentation*, 1972, Vol. 28, pp.11-21.
10. Lesk, M., "Automatic Sense Disambiguation: How to Tell a Pine Cone from and Ice Cream Cone", *Proceedings of the SIGDOC'86 Conference, ACM*, 1986.
11. Mooney, R. J. and Bunescu, R. "Mining Knowledge from Text Using Information Extraction", *SIGKDD Explorations* (special issue on Text Mining and Natural Language Processing), 7, 1 (2005), pp. 3-10.
12. Palakal, M., Stephens, M.; Mukhopadhyay, S., Raje, R., Rhodes, S., "A multi-level text mining method to extract biological relationships" , *Proceedings of the IEEE Computer Society Bioinformatics Conference (CBS2002)*, 14-16 Aug. 2002 Page(s):97 - 108
13. Ponte, J.M. and Croft, W.B., "A Language Modeling Approach to Information Retrieval", *Proceedings of the 21st annual international ACM SIGIR conference on Research and Development in Information Retrieval*.
14. Salton, G., Wu, H., and Yu, C.T., "The measurement of term importance in automatic indexing", *Journal of the American Society for Information Science*, 1981, 32(3), pp.175-186.
15. Sanderson, M. 1994, "Word sense disambiguation and information retrieval", *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, p.142-151, July 03-06, 1994, Dublin, Ireland.
16. Schenker, A., Last, M., Bunke, H., and Kandel, A., "Clustering of Web Documents Using a Graph Model", *In A. Antonacopoulos & J. Hu (Eds.), Web Document Analysis: Challenges and Opportunities*, 2003.
17. Soderland, S., Fisher, D., Aseltine, J., and Lehnert, W., "CRYSTAL: Inducing a Conceptual Dictionary", *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, 1995, pp. 1314-1319.
18. Soderland, S., "Learning Information Extraction rules for Semi-structured and free text", *Machine Learning*, Vol. 34, 1998, pp. 233-272.
19. Stokoe, C. and Tait, J. I. 2004. Towards a Sense Based Document Representation for Information Retrieval, in *Proceedings of the Twelfth Text REtrieval Conference (TREC)*, Gaithersburg M.D.
20. Zhou, X., Han, H., Chankai, I., Prestrud, A., and Brooks, A., "Converting Semi-structured Clinical Medical Records into Information and Knowledge", *Proceeding of The International Workshop on Biomedical Data Engineering (BMDE) in conjunction with the 21st International Conference on Data Engineering (ICDE)*, Tokyo, Japan, April 5-8, 2005.