A System for Finding Biological Entities that Satisfy Certain Conditions from Texts

Wei Zhou
Department of Computer Science
University of Illinois at Chicago
Chicago, IL
wzhou8@uic.edu

Clement Yu
Department of Computer Science
University of Illinois at Chicago
Chicago, IL
yu@cs.uic.edu

Weiyi Meng
Department of Computer Science
Binghamton University
Binghamton, NY
meng@cs.binghamton.edu

ABSTRACT

Finding biological entities (such as genes or proteins) that satisfy certain conditions from texts is an important and challenging task in biomedical information retrieval and text mining. It is essential for many biomedical applications, such as drug discovery which normally requires collecting existing scientific facts from documents. This paper presents an effective IR system for this task, in which 1) domain knowledge is incorporated to improve retrieval effectiveness; 2) query expansion with related concepts on multiple semantic levels is employed; 3) a gene symbol disambiguation technique is implemented. We evaluated these techniques and examined two different concept-based IR models. Experiments based upon the proposed framework yield significant improvement (22% for automatic and 16.7% for non-automatic) over the best reported results of passage retrieval in the Genomics track of TREC 2007.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*retrieval models, query formulation, information filtering*; H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*thesauruses*

General Terms

Algorithms, Performance, Experimentation

Keywords

Document Retrieval, Passage Extraction, Biomedical Documents

1. INTRODUCTION

Information retrieval and text mining systems in the biomedical or genomic domain have become essential tools for researchers and other users. These systems, ranging from ad hoc IR systems (such as PubMed, the best-known biomedical IR system) to advanced text mining systems (such as MedMiner which post-processes documents returned by public search engines, including GeneCards and PubMed), facilitate researchers like biologists in their daily literature search, enable them to collect specific biomedical facts from a large amount of documents, and allow them to find biological entities from texts.

The system presented in this paper is also developed to support biologists' research by addressing one of their important information needs: finding biological entities that satisfy certain conditions from texts. The Genomics track of the Text REtrieval Conference (TREC) provides a common platform to assess the methods and techniques proposed by different groups for biomedical information retrieval. During the past few years, its ad hoc IR task has been designed to meet information needs of biologists in their research. The system introduced in this paper is developed based on the 2007 ad hoc IR task [14], which is more challenging than the task in 2006. The queries in 2006, like "`How does p53 affect apoptosis?`", are asking for relationships between biological objects and biological processes. The objects and processes are explicitly specified in the queries. In this example, the biological object is `p53` and the biological process is `apoptosis`. Systems only need to retrieve those documents (or portions of documents) which mention the objects and processes. It is a typical IR task. In 2007, systems need to identify instances of different types of entities. For example, "`What [TUMOR TYPES] are found in zebrafish?`". In this query, no specific `tumor` is given and a suitable system has to find documents (or portions of documents) which mention certain `tumors` in `zabrafish`.

Two efforts could be potentially effective for this task. Like many other biomedical IR systems, the first effort is to make use of domain knowledge. An obvious use of domain knowledge for this task is to retrieve instances of different entity types from domain ontologies (e.g., obtain all possible instances of `TUMOR TYPES`). The second effort is to use thesauruses to automatically expand the query with related terms (e.g., synonyms). These two efforts are attempted in our system. Besides, a gene symbol disambiguation method is implemented. The system employs a concept-based IR framework in which the similarity between a query and a document is measured on both concept and word levels.

The contributions of this paper are listed as follows:

CIKM'08, October 26–30, 2008, Napa Valley, California, USA.

1. It presents an effective genomic IR system which successfully incorporates domain knowledge and utilizes multiple levels of related terms for query expansion. This system achieves significant improvement (22% for automatic and 16.7% for non-automatic) over the best reported results of passage retrieval in the Genomics track of TREC 2007.

2. Two conceptual IR models are compared in retrieval effectiveness and one of them is found to be more suitable to capture the characteristics of the TREC 2007 queries.

3. A method of finding lexical variants beyond what were previously known is given. Another method is proposed to find related abbreviations whose long-forms[1] contain the query concepts. A simplistic method is also introduced to disambiguate gene symbols. All these three methods are confirmed experimentally to be effective.

This paper is organized as follows: related works are given in the next section. The problem statement is given in section 3. The techniques are introduced in section 4. Section 5 presents the experimental results and we conclude the paper in section 6.

2. RELATED WORKS

In this section, we sketch related works in the areas of Question Answering, biomedical IR systems, query expansion, and incorporating domain knowledge.

Question Answering (QA)

Biological questions in the Genomics track of TREC 2006 and 2007 are formulated as queries. A system is required to find passages (a passage could be a part of a sentence, a sentence, a set of consecutive sentences or a paragraph) of minimum lengths from full-text documents where answers to the questions can be found. These tasks are essentially domain-specific question answering tasks. The difference is that QA systems return short answers (i.e., answer-strings), systems for the tasks of the Genomics track return contexts (i.e., passages), which usually are much longer. Since 1999, the QA track of TREC has continuously focused on developing systems returning answers to natural language questions [6]. Different types of quesions have been investigated. The type of factoid questions, like "`Who founded the House of Chanel?`", asks for facts. Similar type of questions has been proposed in the Genomics track of TREC 2006, for example, "`What is the role of IDE in Alzheimer's disease?`". Another major type of questions in the QA track is the list questions which ask for a list of answer instances, for example, "`What museums have displayed Chanel clothing?`". Questions in the Genomics track of TREC 2007 are all list questions, such as "`What [TUMOR TYPES] are found in zebrafish?`".

Biomedical IR Systems

A number of tools have been developed to assist biomedical scientists in their literature search. A) Some tools create co-occurring networks among genes [16, 18], networks among general biological concepts [4] (e.g., proteins and drugs), or networks among metadata fields [10] (for example, authors). The derived networks give graphical visualization to help users browse the search results by navigating through the networks. B) Some tools allow a user to search for articles using a plain-text natural language question [12], a paragraph of texts [20], a gene/protein sequence [34], or a set of seed PubMed abstracts as the input [13] and then rank the retrieved articles according to their similarities. C) Many other tools post-process the search results of a PubMed query and provide insight and summary of the retrieved literature. The search results can be categorized based on common ontologies, such as Gene Ontology [9] or the Medical Subject Headings (MeSH) hierarchy [33, 31], or a set of manually organized categories [25], allowing a user to browse the articles through the hierarchy. In addition, the search results can be clustered according to their similarities [11], metadata fields, n-grams, or keywords [5, 8, 17, 24, 26]. Also, the search results can be ranked according to journal impact factor and forward referencing [27].

The differences between our system and the above systems are described as follows: First, the queries that our system attempts to answer are more focused: they are asking for a list of biological entities that satisfy certain conditions. Because of this, our system is likely to be more effective for such queries. Second, our system is a passage-retrieval system, which extracts passages from full-text documents, whereas the above systems are document-retrieval systems.

Query Expansion with Related Terms

Many studies used manually-crafted thesauruses or knowledge databases created by text mining systems to automatically expand the query with related terms.

[15, 1] assessed query expansion using the UMLS Metathesaurus, a large database of biomedical terms. Based on a word-statistical retrieval system, [15] used definitions and different types of thesaurus relationships for query expansion and a deteriorated performance was reported. [1] expanded queries with phrases and UMLS concepts determined by the MetaMap, a program which maps biomedical text to UMLS concepts, and no significant improvement was shown.

Many similar studies have been made in the Genomics track of TREC. [2] have shown that query expansion with acronyms and lexical variants of gene symbols produced the biggest improvement in their IR systems, whereas the query expansion with synonyms of gene symbols using gene databases caused deterioration in retrieval performance. [37] used a similar approach for generating lexical variants of gene symbols and reported significant improvements. [39] studied query expansion with five different types of terms, among which the lexical variants produced the biggest improvement in retrieval effectiveness.

[35, 23] utilized WordNet, a database of English words and their lexical relationships developed by Princeton University, for query expansion in a non-biomedical domain. In their studies, queries were expanded using the lexical semantic relations such as synonyms, hypernyms, or hyponyms. Little benefit was shown in [35]. In [23], the sense of a query term is determined using WordNet before their related terms are added into the query and the improvement is significant.

In comparison, we study more levels of related terms in the biomedical domain for query expansion. We utilize an abbreviation database to find additional related concepts, including approximately matched long-forms and those abbreviations whose long-forms contain the query concepts. We carry out comprehensive experiments to look into the effects of different levels of related terms in performance contribution.

[1]For example, `APC/antigen presenting cell` is a pair of abbreviation/long-form.

Incorporating Domain Knowledge

[21] presented a good study of the role of knowledge in the document retrieval of clinical medicine. They have shown that appropriate use of semantic knowledge in a conceptual retrieval framework can yield substantial improvements. [32] also utilized systems/tools, which rely on domain knowledge, to answer questions in clinical practice. Improvement in searching of biomedical abstracts due to knowledge-base processing has been shown. Instead of the domain of clinical medicine, we present a case study of conceptual retrieval in the domain of genomics, where many knowledge resources are also available and can be used to improve the performance of IR systems.

3. PROBLEM STATEMENT

We describe the queries, document collection and the system output in this section.

The query set used in the Genomics track of TREC 2007 consists of 36 questions recently collected from biologists, most of which are related to their latest research. As described in [14], these questions are asking for a list of entities (e.g., genes/proteins) that satisfy certain conditions. A sample question is given below:

`What [GENES] regulate puberty in humans?`

where the type of entities is `GENES` which is contained in the square brackets. The characteristics of the queries are: 1) The first or second word of each query is either `What` or `Which`. It is essentially the same as the LIST question [6], which asks for different instances of a particular type; 2) Domain knowledge is needed to help find the relevant information. For example, for the above query, domain expertise will determine that those genes that regulate puberty in non-human species, such as rat or drosophila (i.e., fruit flies), do not satisfy.

The document collection contains 162,259 Highwire full-text documents in HTML format.

The output of the system is a list of passages ranked in descending order of their similarities to the query (The Genomics track of TREC 2007 only allows up to top 1,000 passages to be returned). Passages must contain one or more instances of the given entity type with supporting text that answers the given question to be considered as relevant. A passage is defined as any span of text that does not include the HTML paragraph tag (i.e., <P> or </P>). A passage could be a part of a sentence, a sentence, a set of consecutive sentences or a paragraph.

4. TECHNIQUES AND METHODS

We approach this problem with a three-step strategy: step 1) The raw query is preprocessed and expanded with related concepts[2] on multiple semantic levels; step 2) top-k ranked paragraphs are retrieved under a conceptual IR framework; step 3) within each paragraph output by step 2, one or more passages are extracted and finally all these passages are ranked according to their similarities. Techniques and methods employed in this strategy are described in the following sections: In section 4.1, we describe how concepts are identified from queries and texts. In section 4.2, we introduce the use of domain knowledge in our system. In section 4.3, we specify five levels of related concepts and describe how they are obtained. In section 4.4, a simple method for gene symbol disambiguation is introduced. In section 4.5, we present our conceptual IR models.

[2]Concepts, in this paper, are defined as those entry terms in the biomedical thesauruses, such as the UMLS Metathesaurus.

4.1 Identifying Concepts in Queries and Texts

We use the query translation functionality of PubMed to extract MeSH terms in a query. This is done by submitting the whole query to PubMed, which will then return a file in which the MeSH terms in the query are labeled. We also look up query terms in Entrez Gene database to find gene names in the query.

Concept identification in the text collection is done in a different way. We use windows of certain sizes to determine whether a phrase concept occurs within a document. If the phrase concept is a gene name, such as `MMS 2`, we require that its component words (`MMS` and `2`) co-occur exactly adjacent to each other and in the given order (`MMS` is on the left of `2`). If the phrase concept is not a gene name, all the component words in that phrase are required to co-occur within a window containing $N = n + (n-1) \times k$ content words, where n is the number of content words in the phrase concept, and k is a small positive constant. For example, in the text "`...Women who are postmenopausal and who have never used hormone replacement therapy have a higher risk of colon, but not rectal, cancer than do women who ...`", `colon` and `cancer` do not appear exactly adjacent to each other. But there is only 1 content word separating them (i.e., "rectal"). As such, a window size of 3 or larger (i.e., $k \geq 1$) will recognize `colon cancer` in this text. An empirically based value of $k = 2$ is used in our system for concept identification in texts. Further systematic optimization of k will be carried out.

4.2 Use of Domain Knowledge

Domain knowledge usually refers to a collection of statements about a certain domain. In our context, as a simple example, a biomedical statement can be that `Hepatitis B virus infects the liver of hominoidae, including humans`, which describes a relation between a virus and a organ. Use of this kind of domain knowledge potentially can improve retrieval effectiveness of biomedical IR systems. In this section, we explain what biomedical domain knowledge is incorporated and how it is utilized. First, we introduce a method for gene/protein species control: if a query is asking for genes/proteins related to a specific species, then genes/proteins related to other species are considered irrelevant. Second, we describe how instances of entities from different resources are obtained.

Gene/protein Species Control

The motivation is that a gene symbol in texts might refer to multiple genes in different species. For example, the gene symbol `prnp` could mean the `prnp` gene of `Homo sapiens` (human), or it could refer to the `prnp` gene of `Rattus norvegicus` (rat). They are called `homologous genes`, which are related via a common ancestral species and retain a similar sequence and function (from NCI Thesaurus). Usually they share common terms for reference. For a biomedical IR system, when a query is asking for genes of a specific species, such as human, those genes of other species are not qualified. Based on this motivation, for those queries asking for genes/proteins, we have the following gene/protein

species control:

CASE 1: if a certain species s is specified in the query, then those passages having genes/proteins of other species but not of s are excluded.

We search for indicative terms in texts to identify the species. For example, `human` and `Homo sapiens` are two indicative terms for the species of human. Note that documents having genes of unknown species (indicative words are not available) are not excluded. But these documents are assigned lower similarity scores than those documents in which the species s is explicitly specified.

CASE 2: it is possible that a query involves genes/proteins of multiple species, for example, 2 species s_1 and s_2. In this case, those passages having genes/proteins of other species but not of s_1 or s_2 are excluded.

For queries in which the species is unknown, this gene/protein species control does not apply.

Compilation of Instances from Thesauruses

As mentioned in the problem statement section, each query requests for a set of instances of a specified entity type. For each entity type, we compile a list of instances from different resources. The main resource that we utilize is the Unified Medical Language System (UMLS) Metathesaurus [22]. The 2006AB version of UMLS Metathesaurus contains information about more than 1.3 million concepts and 6 million unique concept names from more than 100 different source vocabularies. In UMLS, each concept maps to one or more semantic types like `Disease or Syndrome`. We use the mapping between specified entity types and UMLS semantic types introduced in [7]. Their mapping is established by experts from the National Library of Medicine (NLM) and has been demonstrated to be effective for their experiments on the same task. As an example, the entity type `TUMOR TYPES` maps to the following three UMLS semantic types: `Neoplastic Process`, `Pathologic Function`, and `Hazardous or Poisonous Substance`. Concepts that map to any of these three semantic types in the UMLS are considered as instances of `TUMOR TYPES`. Instances are also retrieved from other resources: 1.7 million genes are obtained from Entrez Gene database[3]; 65,381 instances of antibodies are obtained from the Santa Cruz Biotechnology Inc.[4]; 5,668 drug names are obtained from FDA[5]; 375 pathway names are obtained from BioCarta[6] and 325 from KEGG[7].

4.3 Related Concepts

Five levels of related concepts are defined as follows:

Level 1 Synonyms (terms that refer to the same meaning)

Level 2 Hypernyms (more generic terms, one level only)

Level 3 Hyponyms (more specific terms, one level only)

Level 4 Lexical variants (variations of the same concept, such as abbreviations. They are commonly used in the literature, but may not be collected in the thesauruses)

Level 5 Abbreviations whose long-forms contain the query concept (see "Related Abbreviations" of a following subsection for details)

Synonyms, hypernyms, and hyponyms can be obtained from the thesauruses directly if they exist. Note that, for genes, only synonyms are obtained from Entrez Gene since it does not have a hierarchy. Level 4 and level 5 related concepts are obtained as described in the following.

Lexical Variants

Lexical variants of gene symbols are obtained using a method described in [39]. This method is able to automatically generate lexical variants for a gene symbol according to some manually crafted heuristics. It also uses an abbreviation database to retrieve extra lexical variants. We now describe how additional lexical variants beyond those in [39] are obtained through recognizing computationally equivalent long-forms[8] which share the same abbreviation. Long-forms that have the same abbreviation are considered to be computationally equivalent if they satisfy the conditions to be one of the six types as described in Table 1. For example, `human papillomavirus` and `human papillomaviral` have the same abbreviation `HPV` and they are different by a small edit distance [19]. Thus they are considered as lexical variants (or computationally equivalent long-forms). The abbreviations and their long-forms are retrieved from ADAM [38], an abbreviation database which covers frequently used abbreviations and their definitions (or long-forms) within MEDLINE, the authoritative repository of citations from the biomedical literature maintained by NLM. A formal definition of computationally equivalent long-forms with common abbreviation is given as follows:

Definition 1. Given an abbreviation *abbr* and two of its long-forms $long_1$ and $long_2$, the two long-forms are computationally equivalent if, after normalization (including Porter stemming and transforming Greek characters into their corresponding English characters and transforming Roman numerals into Arabic numerals) if necessary, their normalized forms $long_1'$ and $long_2'$ satisfy any of the following conditions:

For type 1 and 2 (see Table 1): $long_1'$ and $long_2'$ are identical after stemming or normalization.

For type 3: $long_1'$ and $long_2'$ are different by at most a small edit-distance t (Determining the threshold of t is discussed following *Definition* 1)

For type 4: $long_1'$ and $long_2'$ are different by at most a small edit-distance after their component words are sorted alphabetically. For example, `amyloid beta protein` vs. `beta amyloid protein` (In this example, the edit distance is 0).

For type 5: the longer string of $long_1'$ and $long_2'$ contains all the component words of the shorter string. The ordering of words in the shorter string is identical to the ordering of matching words in the longer string. For example, `acid sodium citrate dextro` vs. `acid citrate dextro`.

For type 6: Let S and L be the shorter string and the longer string of $long_1'$ and $long_2'$ respectively. L can be split into the left part L_L and right part L_R such that one of the following two conditions satisfies: a) (this condition is to capture pairs like `ag presenting cell` vs. `antigen presenting cell`) let $S_{firstword}$ be the first component word of S and S_R be the remaining part of S. This case requires that: (i) $S_{firstword}$ has 3 or fewer characters; (ii) $L_R = S_R$; (iii) L_L contains all the characters in $S_{firstword}$. In addition, the first character of $S_{firstword}$ is contained in the first

[3]Entrez Gene: http://www.ncbi.nlm.nih.gov/sites/entrez?db=gene

[4]Santa Cruz Biotechnology Inc. http://www.scbt.com

[5]FDA: http://www.fda.gov/cder/drugsatfda/datafiles/default.html

[6]BioCarta: http://www.biocarta.com/genes/allPathways.asp

[7]KEGG: ftp://ftp.genome.ad.jp/pub/kegg/pahway/

[8]It should be noted that the determination whether two long forms are "actually equivalent" can only be done manually by an domain expert.

Table 1: Different types of lexical variants

Type 1:	Identical after stemming
APC::	"antigen presenting cell" ≈ "antigen presented cell"
Type 2:	Identical after transforming
CK2	"Casein Kinase II" ≈ "Casein Kinase 2"
Type 3:	Different by a small edit distance
HPV:	"Human papillomavirus" ≈ "Human papillomaviral"
Type 4:	Different ordering
Abeta:	"amyloid beta protein" ≈ "beta amyloid protein"
Type 5:	Extra words
ACD:	"acid citrate dextro" ≈ "acid sodium citrate dextro"
Type 6:	Internal abbreviations
APC:	"ag presenting cell" ≈ "antigen presenting cell"

component word of L_L and the last character of $S_{firstword}$ is contained in the last component word of L_L. The ordering of characters in $S_{firstword}$ is identical to the ordering of matching characters in L_L; b) (this condition is to capture pairs like `Activated Protein C` vs. `Activated PC`) let $S_{lastword}$ be the last component word of S and S_L be the remaining part of S. This case requires that: (i) $S_{lastword}$ has 3 or fewer characters; (ii) $L_L = S_L$; (iii) L_R contains all the characters in $S_{lastword}$. In addition, the first character of $S_{lastword}$ is contained in the first component word of L_R and the last character of $S_{lastword}$ is contained in the last component word of L_R. The ordering of characters in $S_{lastword}$ is identical to the ordering of matching characters in L_R. ■

For types 3 and 4, a threshold of edit-distance, t, is set at 2. To determine the optimal t, we varied the edit-distance from 1 to 4. Thus 4 sets of candidate pairs were generated, S_1, S_2, S_3, and S_4. For each $i = 1, 2, 3$, and 4, 20 candidate pairs were randomly selected from S_i. These 20 pairs were then manually examined. The numbers of pairs that are actually computationally equivalent were 18, 16, 3, and 0 for $i = 1, 2, 3$, and 4, respectively, which indicates that a large portion of non-equivalent pairs were included when the edit-distance is bigger than 2. Therefore, a threshold of 2 is set for t, which means $long'_1$ and $long'_2$ are determined to be computationally equivalent if their edit-distance is $\leq t(= 2)$.

Related Abbreviations

The **motivation** of this type of related concepts is explained with the following example. Suppose we have a query involving a concept `lung cancer`. It is possible that a document has a definition for an abbreviation `SCLC` which stands for `small cell lung cancer` in a paragraph of a full-text document and in later paragraphs, `SCLC`, instead of `small cell lung cancer`, is used. In this case, we will not capture those passages in later paragraphs in which `SCLC` is used but `lung cancer` is not explicitly written. One solution to this problem is given as follows:

Step 1: Given a query concept, find all those abbreviations whose long-forms contain the query concept. The ADAM database [38] was used for this step. For the above example, more than 8 different abbreviations whose long-forms contain `lung cancer` are retrieved, including `SCLC (small cell lung cancer)`, `NSCLC (non-small cell lung cancer)`, and `LCSS (lung cancer symptom scale)`.

Step 2: Related abbreviations obtained by step 1 are added into the query as alternatives of that query concept. A passage having one of these related abbreviations is considered to have the original query concept if the long-form of that abbreviation in the corresponding full-text document contains the query concept. An abbreviation identification program [3] is used to extract abbreviations/long-forms in texts.

The above strategy of query expansion with such type of related abbreviations is new and it applies to passage-level information retrieval when abbreviations appear in passages in which their long-forms are not present.

4.4 Gene Symbol Disambiguation

Many gene symbols are ambiguous, not only because a gene symbol may refer to multiple different genes, but it may have one or more non-gene meanings. Many research efforts have been conducted for gene symbol disambiguation, including a supervised-learning method [28] in which a training set is automatically generated for each human gene, a thesaurus-based method [30] in which a reference description based on either its annotations or MEDLINE abstracts is created for each human gene, and a general method [36] in which Entrez Gene is used to create a profile for each gene sense. All these existing methods need a lot of efforts to create either a training set or a profile for each ambiguous gene symbol. In this paper, for simplicity and efficiency reasons, we develop a set of simple rules to disambiguate gene symbols:

RULE 1: Suppose a gene symbol g is an abbreviation in document d. If its long-form in d is a gene, then g has a gene meaning in d, else it has a non-gene meaning in d.

As an example, suppose the ambiguous gene symbol `NOD` is an abbreviation in a document and its long-form in the document is `non-obese diabetic` which is not a gene name. Thus the gene symbol `NOD` has a non-gene meaning in this document.

RULE 2: Suppose term g is a gene symbol and in document d, g' is an abbreviation and g is contained in g'. If in document d, g' has a non-gene long-form but there is no long-form for g, then g has a non-gene meaning in the document.

This rule is an extension of RULE 1. For example, suppose the gene symbol `HIV` occurs in a document in which `HIV 1` is an abbreviation for `human immunodeficiency virus type 1` which is not a gene name. Then the gene symbol `HIV` in this document has a non-gene meaning.

RULE 3: Suppose word w is a gene symbol and w is also a word in WordNet. If in document d, w is adjacent to any of the following words: `gene, mutant, mutation, oncogene, mRNA, DNA, cDNA, target, homologue, expressed, repressed, inhibit`, then w has a gene meaning in d, else it has a non-gene meaning in d.

Some gene symbols are also ordinary English words, such as `cat`, `kit`, or `male`. If a document explicitly mentions a phrase such as "`gene` X" or "X `oncogene`", then X has a gene meaning in this document, otherwise, we assume that it is likely to have a non-gene meaning. To find those gene indicative words, we choose a set of common gene symbols and extract those content words that occur immediately before or after them in the MEDLINE abstracts and rank these words by their document frequencies. The above 12 most indicative words having high frequencies are selected by an expert with a biology background.

4.5 Conceptual IR Models

We now discuss our conceptual IR models. We consider that each query consists of two parts, target and qualification. Target refers to instances of a certain entity type and qualification refers to the condition that the target needs to satisfy in order to be qualified as an answer to the query. For example, the target and the qualification of the query "`What [ANTIBODIES] have been used to detect protein TLR4?`" are "`instances of ANTIBODIES`" and "`have been used to detect protein TLR4`", respectively. The similarity of a document to a query is measured by the degree the document contains one or more target instances and satisfies the qualification.

A query q involving an entity type of g can be written as:

$$q = [g, < c_1, c_2, \ldots, c_m >, < w_1, w_2, \ldots, w_n >] \quad (1)$$

where g is the entity type in q, $< c_1, c_2, \ldots, c_m >$ is a vector of qualification concepts, $< w_1, w_2, \ldots, w_n >$ is a vector of query content words. In the above example, g is `ANTIBODIES`, $< c_1, c_2, \ldots, c_m >$ is <`protein TLR4`>, and $< w_1, w_2, \ldots, w_n >$ is <`antibodies, detect, protein, TLR4`> (`what, have, been, used, to` are stop words.). Suppose for target g, we can retrieve k instances from domain ontologies denoted by $< g_1, g_2, \ldots, g_k >$. It is possible that documents contain different instances of type g, which raises an interesting issue regarding whether instances of the same entity type should be differentiated. We explore two different models to investigate this issue. One model assigns weights to instances in $< g_1, g_2, \ldots, g_k >$ according to their document frequencies. The other model does not differentiate documents which have one or more instances of the entity type. These two models are described below.

Model 1: Differentiate Instances

The similarity between query q and document d is measured on two levels: concept level and word level. The concept-level similarity is obtained by matching target instances and qualification concepts, while the word-level similarity is obtained by matching content words.

$$sim(q, d) = (sim(q, d)_{concept}, sim(q, d)_{word}) \quad (2)$$

In model 1, each instance g_i in $< g_1, g_2, \ldots, g_k >$, is assigned a weight $wt(g_i)$ using the IDF (inverse document frequency weight) of g_i. Each concept c_i in the qualification also has a weight, $wt(c_i)$, which is equal to IDF of c_i. Two values, one from target and the other from qualification, are computed by matching target instances and qualification concepts against concepts found in document d. The final concept similarity is a linear combination of these two values:

$$\begin{aligned} sim(q, d)_{concept} &= \alpha \times MAX\{wt(g_i) | g_i \in d\} \\ &\quad + (1 - \alpha) \times \textstyle\sum_{c_i \in d} wt(c_i) \end{aligned}$$

where α is a tuning parameter ($\alpha = 0.25$ is used in our experiments; the tuning of α is discussed in the section of EXPERIMENTAL RESULTS), $MAX\{wt(g_i)|g_i \in d\}$ is the value from target, $\sum_{c_i \in d} wt(c_i)$ is the value from qualification. The word-level similarity is computed using Okapi [29]. Given two documents d_1 and d_2, we have $sim(q, d_1) > sim(q, d_2)$ or d_1 will be ranked higher than d_2, with respect to the same query, if either:

$$\begin{aligned} &1) \quad sim(q, d_1)_{concept} > sim(q, d_2)_{concept} \quad OR \\ &2) \quad sim(q, d_1)_{concept} = sim(q, d_2)_{concept} \\ &\quad \& \quad sim(q, d_1)_{word} > sim(q, d_2)_{word} \end{aligned}$$

This conceptual IR model emphasizes the similarity on the concept level. More precisely, documents are ranked in descending order of the concept-level similarity. Only when documents have the same similarity in concepts, then the similarities in words are used to break ties.

Model 2: Equally Weight Target Instances

While model 1 combines concept similarity of target and that of qualification into one concept similarity, model 2 both emphasizes and de-emphasizes the target. In model 2, it is critical for a document to have a target instance in order to be retrieved (i.e., emphasize the target). However, documents having one or more target instances are differentiated not by the target instances, but by their degrees of satisfying the qualification. In other words, all documents having at least one g_i in $< g_1, g_2, \ldots, g_k >$ receive the same similarity value with respect to the target (i.e., de-emphasize the target). The similarity between query q and document d is measured on three levels: target concept-level, qualification concept-level and word-level. The value of the target concept-level similarity is either 0 or 1 depending on whether d has any target instance. The qualification concept-level similarity is obtained by matching qualification concepts, while the word-level similarity is obtained by matching content words.

$$sim(q, d) = (sim(q, d)_{target}, sim(q, d)_{qlf}, sim(q, d)_{word})$$

Given two documents d_1 and d_2, we have $sim(q, d_1) > sim(q, d_2)$ or d_1 will be ranked higher than d_2, with respect to the same query, if either:

$$\begin{aligned} &1) \quad sim(q, d_1)_{target} > sim(q, d_2)_{target} \quad OR \\ &2) \quad sim(q, d_1)_{target} = sim(q, d_2)_{target} \\ &\quad \& \quad sim(q, d_1)_{qlf} > sim(q, d_2)_{qlf} \quad OR \\ &3) \quad sim(q, d_1)_{target} = sim(q, d_2)_{target} \\ &\quad \& \quad sim(q, d_1)_{qlf} = sim(q, d_2)_{qlf} \\ &\quad \& \quad sim(q, d_1)_{word} > sim(q, d_2)_{word} \end{aligned}$$

In this model, documents having at least one target instance will be ranked higher than documents having no target instances at all. For those documents having at least one target instance, the qualification concept-level similarity will be used next to break the tie. If two documents continue to have a tie on the qualification concept-level similarity, then the word-level similarity will be used to break the tie.

Both model 1 and model 2 are different from the model used in [39]. TREC 2006 genomics queries have two components, biological objects (v_1) and biological process (v_2). These two components are considered equally important in [39]. v_i, $i = 1, 2$, may involve multiple concepts and the $sim(q, d)_{concept}$ of v_i captures the degree of v_i that document d has covered by $\alpha_i = \frac{\sum_{c \in d \& c \in v_i} IDF_c}{\sum_{c \in v_i} IDF_c}$. TREC 2007 genomics queries also have two components, the entity type and qualification. As described above, in model 1, the qualification component is considered more important than the entity type ($\alpha = 0.25$, $1 - \alpha = 0.75$), whereas in model 2,

the component of entity type is considered more important than the qualification.

Query Expansion with Related Concepts

Given a concept c, a vector u is derived by incorporating its related concepts: $u =< c, u_1, u_2 >$ where u_1 is a vector of its synonyms, hyponyms, lexical variants, and related abbreviations and u_2 is a vector of its hypernyms. An occurrence of any concept in u_1 will be counted as an occurrence of c. But a document in which only some hypernym of c occurs will receive a portion (β varies from 0.55 to 1; the tuning of β is discussed in the section of EXPERIMENTAL RESULTS.) of the weight on concept c, assuming that the original concept and its synonyms, hyponyms, lexical variants or related abbreviations have a higher priority than its hypernyms. Whenever a document has occurrences of c or one or more occurrences of $c' \in u_1 \cup u_2$, it is given a concept similarity which is the maximum of the weights of the matching occurrences. This is more or less equivalent to applying the Boolean "OR" operator.

5. EXPERIMENTAL RESULTS

Data Sets and Evaluation Metrics

Our experiments are based on the Genomics track of TREC 2007. The document collection contains 162,259 full-text documents from 49 Highwire biomedical journals. The set of queries consists of 36 queries recently collected from biologists.

The performance is measured on three different levels (passage, aspect, and document) to assess how well the question is answered from different perspectives: **Passage MAP**: it is a character-based precision measure (known as passage2 MAP). The ideal passages are those having all query concepts and are as short as possible. **Aspect MAP**: This measure indicates how comprehensive the question is answered. **Document MAP**: This is a standard IR measure. For a set of queries, the mean of the average precision for all queries is the MAP of that IR system. Details of these evaluation metrics can be found in [14].

The Wilcoxon signed-rank test is employed to determine the statistical significance of one result compared to another result. In Table 2, statistically significant improvements (at the 5% level) are marked with an asterisk. A value of 95% is used for the parameter β for query expansion in the experiments. We use the passage extraction strategy introduced in [39] which assumes that an optimal passage in a paragraph should have all the query concepts that the whole paragraph has. In addition, such a passage should have a high density of query concepts. This passage extraction method is able to extract multiple passages from a single paragraph.

Impact of Domain Knowledge

To evaluate the hypothesis that incorporating domain knowledge improves retrieval effectiveness, an initial baseline is established based on model 1 (indicated by BS1 in Table 2) without using any domain knowledge (i.e., gene/protein species control is not applied and instances retrieved from resources are not used). Another run, BS1+K(M1), based on the same model (model 1) is performed by incorporating domain knowledge as described in section 4.2. Table 2 gives the experimental results, which clearly demonstrate that incorporation of domain knowledge affects most of the queries and yields statistically significant improvement on performance across all three measures. Note that the strategy of gene/protein species control (see section 4.3) only applies to queries asking for genes/proteins of specific species. Among the 36 queries, only 2 queries are affected. For these two queries, one involves only one species and the species control strategy yields significant improvement (+54.6% on passage, +28.8% on aspect, and +36.2% on document. see Table 3.a). The other query involves two different species and no significant impact is observed. In Table 3, the MAPs are for queries which are affected by one of the strategies (species control, etc.) and not for the entire set of queries. We can also see from Table 2 that another run, BS1+K(M2), in which model 2 is applied achieves better performance than BS1+K(M1) on all three measures. It suggests that model 2 is more appropriate for the queries in TREC 2007 than model 1.

Impact of Different Related Concepts

A series of experiments are performed to examine how each level of related concepts contributes to the retrieval performance. The BS1+K(M2) is used as the new baseline (BS2). Then six runs are conducted by adding each individual level of related concepts (two runs for the level of lexical variants). We also conduct a run by adding all levels of related concepts. We find that query expansion with any of the five levels of related concepts improves the performance. The biggest improvement comes from the lexical variants [see the run BS2+VAR2 (all lexical variants)], which is consistent with the results reported in [2, 39]. A separate run, BS2+VAR1, is conducted to investigate the impact of lexical variants obtained through recognizing computationally equivalent long-forms (see section 4.3). These are the variants which have not been investigated by other researchers. Some queries are affected by this level of query expansion, others are not. Further analysis shows that for those affected queries, the impact is significant (+36% on passage, +19.5% on aspect, and +24.6% on document, see Table 3.b).

In the run BS2+ABBR, which is to investigate the impact of related abbreviations whose long-forms contain query concepts, 15 queries are affected. For these 15 queries (Table 3.c), the impact is significant (+13.9% on passage, +7.7% on aspect, and +10.8% on document.), which indicates that some passages of a document use related abbreviations, such as `SCLC` (stands for `small cell lung cancer`) with respect to `lung cancer`, instead of the original query concept. Further analysis shows that among these 15 affected queries (i.e., related abbreviations are added into the queries), there is improvement in retrieval effectiveness for 9 queries. For the remaining 6 queries, neither deterioration or improvement is observed. This is because those added related abbreviations for these 6 queries are not defined in the document collection. For example, `UCAD` (stands for `unstable coronary artery disease`) is a related abbreviation for `coronary artery disease` in one of the 6 queries. However, this abbreviation/long-form pair is not defined in the given document collection. We can also see from Table 2 that synonyms provide the second biggest improvement. Hypernyms and hyponyms provide similar degrees of improvement. The overall performance is an accumulative result of adding different levels of related concepts and it is better than any individual addition. It is clearly shown that the performance is significantly improved (+67.1% on passage, +43.7% on aspect, and +32.3% on document) when the related concepts are added. Although it is not explicitly shown in the tables,

Table 2: Impact of domain knowledge

Run	Passage		Aspect		Document	
	MAP	Impv qs#	MAP(Impv %)	Impv qs#	MAP	Impv qs#
BS1	0.041	N/A	0.147	N/A	0.196	N/A
BS1+K(M1)	0.062(+51.2%)*	27	0.188(+27.9%)*	26	0.236(+20.4%)*	25
BS1+K(M2)	0.076(+85.4%)*	30	0.202(+37.4%)*	26	0.254(+29.6%)*	28
BS2[a]+SYNO	0.091(+20.2%)*	11	0.219(+8.6%)*	12	0.281(+10.5%)*	10
BS2+HYPE	0.079(+3.8%)	5	0.203 (+0.5%)	4	0.256 (+0.8%)	3
BS2+HYPO	0.082(+8.4%)	9	0.197 (-2.4%)	3	0.258 (+1.4%)	6
BS2+VAR1[b]	0.085(+12.0%)	9	0.215 (+6.5%)	8	0.275 (+8.2%)	10
BS2+VAR2[c]	0.114(+50.6%)*	18	0.280 (+38.7%)*	17	0.307 (+20.8%)*	16
BS2+ABBR[d]	0.080(+5.8%)	9	0.208 (+3.2%)	8	0.265 (+4.5%)	8
BS2+ALL	**0.127(+67.1%)***	24	**0.289 (+43.7%)***	21	**0.336 (+32.3%)***	20
BS3+GSD[e]	0.134 (+5.5%)	8	0.301 (+4.2%)	8	0.354 (+5.4%)	10

[a] BS2=BS1+K(M2) [b] VAR1=Equivalent long-forms [c] VAR2=All types of lexical variants
[d] ABBR=Related abbreviations [e] BS3+GSD=BS2+ALL+Gene symbol disambiguation

different levels of related concepts affect different subsets of queries. More specifically, each of these types (with the exception of "the lexical variants" which affects a large number of queries) affects only a few queries. But for those affected queries, their improvement is significant. As a consequence, the accumulative improvement is very significant.

Table 3: Impact of query expansion for AFFECTED QUERIES only: a. 2 queries are affected by query expansion using genes/proteins species control; b. 12 queries are affected by query expansion using computationally equivalent long-forms; c. 15 queries are affected by query expansion using related abbreviations; d. 11 queries are affected by gene symbol disambiguation.

Type	Run	Psg MAP	Asp MAP	Doc MAP
a. spec. ctrl.	BS1	0.038	0.145	0.190
	BS1+K(M1)	0.059	0.187	0.258
	Impv	+54.6%	+28.8%	+36.2%
b. VAR1	BS2	0.083	0.224	0.269
	BS2+VAR1	0.113	0.268	0.335
	Impv	+36.0%	+19.5%	+24.6%
c. ABBR	BS2	0.081	0.212	0.263
	BS2+ABBR	0.092	0.228	0.291
	Impv	+13.9%	+7.7%	+10.8%
d. GSD	BS2+ALL	0.120	0.250	0.286
	BS3+GSD	0.144	0.284	0.338
	Impv	+20.0%	+13.7%	+18.2%

Impact of Gene Symbol Disambiguation

Using the BS2+ALL as a new baseline (BS3), the contribution of gene symbol disambiguation is given in the run of BS3+GSD. This method only applies to those queries asking for GENES. Among the 36 queries, there are 11 such queries. As shown in Table 2, the performance of most of these 11 queries is improved (8 on passage, 8 on aspect, and 10 on document). For these 11 affected queries, the impact is significant (+20.0% on passage, +13.7% on aspect, and +18.2% on document. see Table 3.d). Further analysis shows that for the other 3 queries that show no improvement, although some passages that have gene symbols of non-gene meanings were ranked lower by gene symbol disambiguation, their rankings were below the retrieved relevant passages before disambiguation and therefore there is no improvement or deterioration.

Table 4: Comparison with best-reported results

	Psg MAP	Asp MAP	Doc MAP
Best reported(auto)	0.1097	0.2494	0.3105
Best reported(non-auto)	0.1148	0.2631	0.3286
Our result(auto)	0.1340	0.3010	0.3543

Comparison with Best Reported Results

In Table 4, we compare our result with the best results reported in the Genomics track of TREC 2007 [14]. The improvement of our result over the best reported results is significant (22% for automatic and 16.7% for non-automatic in passage retrieval). A system is automatic if a user is allowed to enter the query only, without making any changes to the query or any interaction with the system; otherwise it is non-automatic. Our system is an automatic system.

Tuning Parameters

Besides the above comparison experiments, we also test the parameter sensibility of our model and tune those model parameters to get the optimal results. There are mainly two parameters in our model. The first parameter is the portion (β) of the weight we assign to a concept when its hypernym occurs in a document. The second parameter is the factor α that we use to combine the similarities from target and qualification when we calculate the concept similarity $sim(q,d)_{concept}$ in Model 1. Therefore we have done two sets of experiments to tune these two parameters, respectively.

1. Tuning hypernym weight parameter β. We use BS2+HYPE as our experiment setting and test the system performance with varying β. The result is shown in Figure 1(a). In this set of experiments, we find that the best performance (i.e., the document MAP in our experiments) is obtained when β is around 0.95.

2. Tuning α. We use BS1+K(M1) as our experiment setting and test the performance of Model 1 with varing α. The result is shown in Figure 1(b). In this set of experiments, we find that the best performance is obtained when α is around 0.25.

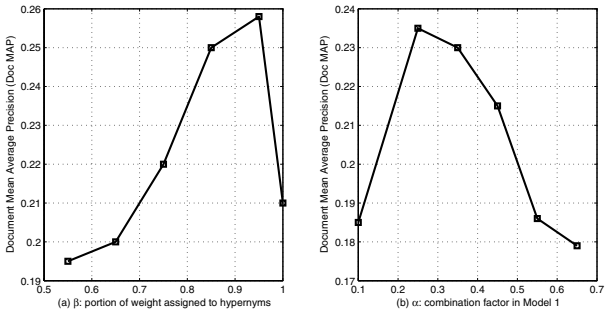

Figure 1: Model parameter sensibility testing results based on document mean average precision (Doc MAP) (the y-axis)

6. DISCUSSION AND CONCLUSION

This paper is a follow-up work of our research on biomedical IR reported in [39]. We now describe the new contributions made in this paper beyond [39]. **1)** We have investigated two new conceptual IR models based on the characteristics of TREC 2007 genomics queries. The difference between these two models and the model used in [39] are explicitly discussed in section 4.5. **2)** TREC 2007 genomics queries are list questions, whereas queries in TREC 2006 are not. Answering these list questions is challenging and requires the use of more biomedical domain knowledge. One of the important steps to build our system is to compile target instances from several resources. The experimental results have indicated that utilizing these resources to retrieve target instances is crucial to the success of such a system. **3)** In [39] we found that the query expansion with lexical variants produced the biggest improvement of performance. As such, more efforts have been made in this paper to find lexcial variants of concepts beyond those in [39]. This includes recognizing computationally equivalent long-forms using abbreviation databases and finding related abbreviations to improve passage-level retrieval. Their impact is shown in Table 2 (BS2+VAR1 and BS2+ABBR, respectively). **4)** Two methods are used to handle gene symbols. One is the species control (see section 4.2) which is to resolve the situation that a gene symbol in texts might refer to multiple genes in different species. The other method is related to disambiguation of gene symbols in general (see section 4.4).

Several techniques or methods are examined in this paper, such as query expansion using knowledge resources. Although some of these techniques seem similar to previously published ones (see RELATED WORKS), they are actually quite different in details. For example, in our query expansion process, for each query concept, its related terms are added as its alternatives through the Boolean operator **OR**. Whereas, in the query expansion process investigated in [15], a query concept, after its related terms are added, becomes "a bag of content words". Given a query having multiple concepts, e.g., "`what is the role of gene prnp in the mad cow disease?`", consider two documents d_1 and d_2, where d_1 does not contain the concept `prnp` or any of its related concepts but has many occurrences of `mad cow disease` and/or its related concepts and d_2 has one occurrence of `prnp` or its related concepts and one occurrence of `mad cow disease` or its related concept. It is likely that d_1 is irrelevant and d_2 is relevant to the query. Traditional term similarity functions, such as the IR system used in [15], will assign higher term similarities to d_1 than d_2 and therefore rank d_1 higher than d_2. This might be the reason why a deteriorated performance was reported for their query expansion. In contrast, our conceptual similarity function will assign higher conceptual similarity to d_2 than d_1 and therefore rank d_2 higher. The fact that the document has many occurrences of `mad cow disease` and/or its related terms will only contribute to its secondary word-level similarity $sim(q,d)_{word}$. As another example, due to ambiguity of the query terms that have different meanings in different contexts, little benefit has been shown in [35] when query expansion was conducted using WordNet. Whereas, in the biomedical domain, this kind of ambiguity of query terms is relatively less frequent, because, although the abbreviations are highly ambiguous, general biomedical concepts usually have only one meaning in the UMLS Metathesaurus, whereas a term in WordNet usually has multiple meanings (represented as synsets in WordNet). The above two examples show that even though some techniques used in our system (such as query expansion explained above) have already been studied, they have been applied differently and proved to be effective in our system.

In summary, we propose a system for finding biological entities (such as genes and proteins) that satisfy certain conditions in texts. We incorporate domain knowledge and study five different levels of related concepts for query expansion (i.e., synonyms, hypernyms, hyponyms, lexical variants, and related abbreviations) and examine their effects on retrieval effectiveness. We evaluate a technique for gene symbol disambiguation. Experimental results have shown that our methods and techniques implemented under a concept model yield significant improvements (22% for automatic and 16.7% for non-automatic) over the best known results of passage retrieval in the Genomics track of TREC 2007. We also compare two concept models and show that one of them is more appropriate to process the queries in TREC 2007. We describe our future work as follows.

1. We will improve the quality of target instances retrieved from different resources. (Some instances are very common terms and they are not "real" instances of their corresponding types. For example, `brain` is an instance of `disease or syndrome` retrieved from the UMLS).

2. In addition, our current method for gene symbol disambiguation is simplistic. It can not be applied when no long-forms or indicative words of the gene symbol are available in the document since the first and second rules use the long-form identified in texts to disambiguate the gene symbol (i.e., the abbreviation) and the third rule uses indicative words to determine whether the gene symbol has a gene meaning or not. More advanced techniques for gene symbol disambiguation will be explored. For example, our current RULE 3 does not consider the cases like: "`genes` X `and` Y" in which X is adjacent to the indicative word `gene`, but Y is 3 words away. Our RULE 3 will miss such cases.

3. Our system recognizes query concepts within each sentence of a paragraph. But we observed some cases like: "`Genetic Creutzfeldt-Jakob disease is associated with pathogenic variations in the` `PRNP gene`. `This gene` `consists of two exons and located on chromosome 20pter-p12.`", in which the `This gene` in the

second sentence actually refers to PRNP gene. Our concept recognition method will not be able to associate This gene with PRNP gene. Since our passage extraction algorithm identifies shortest passages from a paragraph according to the distribution of query concepts in all the sentences of the paragraph, it is essential to recognize query concepts in each sentence.

4. As indicated by our experimental results, BS1 *vs.* BS1+K(M1), incorporation of domain knowledge affects most of the queries and yields statistically significant improvement on performance across all three measures. With more and more biomedical knowledge being encoded into ontologies, the major challenge in our future work is how to utilize these biomedical domain ontologies more appropriately. We also plan to perform more experiments on additional gold standard corpuses other than the TREC collection to further test and improve our system.

7. ACKNOWLEDGMENTS

This research is supported in part by NSF grant IIS-0738652 and NSF grant IIS-0738727.

8. REFERENCES

[1] A. R. Aronson and T. C. Rindflesch. Query expansion using the umls metathesaurus. In *Proc AMIA Annu Fall Symp.*, pages 485–489. American Medical Informatics Association, Oct. 1997.

[2] S. buttcher, C. L. A. Clarke, and G. V. Cormack. Domain-specific synonym expansion and validation for biomedical information retrieval. In *the Thirteenth Text REtrieval Conference (TREC 2004)*. National Institute of Standards and Technology, November 2004.

[3] J. T. Chang, H. Schütze, and R. B. Altman. Creating an online dictionary of abbreviations from medline. *J Am Med Inform Assoc.*, 9(6):612–620, November 2002.

[4] H. Chen and B. M. Sharp. Content-rich biological network constructed by mining pubmed abstracts. *BMC Bioinformatics*, 5(147), October 2004.

[5] ClusterMed. Vivísimo clustermed. http://clustermed.info/.

[6] H. T. Dang, D. Kelly, and J. Lin. Overview of the trec 2007 question answering track. In *the Sixteenth Text REtrieval Conference (TREC 2007)*. National Institute of Standards and Technology, November 2007.

[7] D. Demner-Fushman, S. M. Humphrey, N. C. Ide, R. F. Loane, J. G. Mork, M. E. Ruiz, P. Ruch, L. H. Smith, J. W. Wilbur, and A. R. Aronson. Combining resources to find answers to biomedical questions. In *the Sixteenth Text REtrieval Conference (TREC 2007)*. National Institute of Standards and Technology, November 2007.

[8] A. Divoli and T. K. Attwood. Bioie: Extracting informative sentences from the biomedical literature. *Bioinformatics*, 33(9):2138–2139, February 2005.

[9] A. Doms and M. Schroeder. Gopubmed: Exploring pubmed with the gene ontology. *Nucleic Acids Res.*, 21(Web Server issue):W783–W786, April 2005.

[10] S. M. Douglas, G. T. Montelione, and M. Gerstein. Pubnet: a flexible system for visualizing literature derived networks. *Genome Biol.*, 6(9):R80, July 2005.

[11] A. D. Eaton. Hubmed: a web-based biomedical literature search interface. *Nucleic Acids Res.*, 34(Web Server issue):W745–W747, January 2006.

[12] P. Fontelo, F. Liu, and M. Ackerman. askmedline: a free-text, natural language query tool for medline/pubmed. *BMC Medical Informatics and Decision Making*, 5(5), March 2005.

[13] T. Goetz and C.-W. von der Lieth. Pubfinder: a tool for improving retrieval rate of relevant pubmed abstracts. *Nucleic Acids Res.*, 33(Web Server issue):W774–W778, July 2005.

[14] W. Hersh, A. Cohen, L. Ruslen, and P. Roberts. Trec 2007 genomics track overview. In *the Sixteenth Text REtrieval Conference (TREC 2007)*. National Institute of Standards and Technology, November 2007.

[15] W. Hersh, S. Price, and L. Donohoe. Assessing thesaurus-based query expansion using the umls metathesaurus. In *Proc AMIA Annu Fall Symp.*, pages 344–348. American Medical Informatics Association, November 2000.

[16] R. Hoffmann and A. Valencia. A gene network for navigating the literature. *Nature Genetics*, 36(7):664–664, July 2004.

[17] ISI-knowledge. Isi knowledge. http://isiknowledge.com/.

[18] T.-K. Jenssen, A. Lagreid, J. Komorowski, and E. Hovig. A literature network of human genes for high-throughput analysis of gene expression. *Nature Genetics*, 28(1):21–28, May 2001.

[19] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707–710, 1966.

[20] J. Lewis, S. Ossowski, J. Hicks, M. Errami, and H. R. Garner. Text similarity: an alternative way to search medline. *Bioinformatics*, 22(18):2298–2304, July 2006.

[21] J. Lin and D. Demner-Fushman. The role of knowledge in conceptual retrieval: A study in the domain of clinical medicine. In *SIGIR2006*, pages 99–106, July 2006.

[22] D. A. B. Lindberg, B. L. Humphreys, and A. T. McCray. The unified medical language system. *Methods of Information in Medicine*, 32(4):281–291, August 1993.

[23] S. Liu, F. Liu, and C. Yu. An effective approach to document retrieval via utilizing wordnet and recognizing phrases. In *SIGIR2004*, pages 266–272, July 2004.

[24] U. Mudunuri, R. Stephens, D. Bruining, D. Liu, and F. J. Lebeda. botxminer: mining biomedical literature with a new web-based application. *Nucleic Acids Res.*, 34(Web Server issue):W748–W752, March 2006.

[25] H.-M. Muller, E. E. Kenny, and P. W. Sternberg. Textpresso: an ontology-based information retrieval and extraction system for biological literature. *PLoS Biol.*, 2(11):e309, Nov. 2004.

[26] C. Perez-Iratxeta, P. Borka, and M. A. Andrade. Xplormed: a tool for exploring medline abstracts. *Trends Biochem Sci.*, 26(9):573–575, September 2001.

[27] M. V. Plikus, Z. Zhang, and C.-M. Chuong. Pubfocus: semantic medline/pubmed citations analytics through integration of controlled biomedical dictionaries and ranking algorithm. *BMC Bioinformatics*, 7(424), October 2006.

[28] R. M. Podowski, J. G. Cleary, N. T. Goncharoff, G. Amoutzias, and W. S. Hayes. Azure, a scalable system for automated term disambiguation of gene and protein names. In *the 2004 IEEE Computational Systems Bioinformatics Conference*, pages 415–424. IEEE, August 2004.

[29] S. E. Robertson and S. Walker. Okapi/keenbow at trec-8. In *the Eighth Text REtrieval Conference (TREC 2007)*. National Institute of Standards and Technology, November 2000.

[30] B. J. A. Schijvenaars, B. Mons, M. Weeber, M. J. Schuemie, E. M. van Mulligen, H. M. Wain, and J. A. Kors. Thesaurus-based disambiguation of gene symbols. *BMC Bioinformatics*, 6(149), October 2005.

[31] N. R. Smalheiser, W. Zhou, and V. I. Torvik. Anne o'tate: A tool to support user-driven summarization, drill-down and browsing of pubmed search results. *Journal of Biomedical Discovery and Collaboration*, 3(2), February 2008.

[32] C. A. Sneiderman, D. Demner-Fushman, M. Fiszman, N. C. Ide, and T. C. Rindflesch. Knowledge-based methods to help clinicians find answers in medline. *Journal of American Medical Information Assoc.*, 14(6):772–780, July 2007.

[33] H. Tenner, G. R. Thurnayr, and R. Thurmayr. Data mining with meva in medline. In *the 4th International Symposium on Medical Data Analysis (ISMDA 2003)*, pages 39–46, October 2003.

[34] Q. Tu, H. Tang, and D. Ding. Medblast: searching articles related to a biological sequence. *Bioinformatics*, 20(1):75–77, June 2004.

[35] E. M. Voorhees. Query expansion using lexical-semantic relations. In *SIGIR1994*, pages 61–69, July 1994.

[36] H. Xu, J.-W. Fan, G. Hripcsak, E. A. Mendonça, M. Markatou, and C. Friedman. Gene symbol disambiguation using knowledge-based profiles. *Bioinformatics*, 23(8):1015–1022, February 2007.

[37] M. Zhong and X. Huang. Concept-based biomedical text retrieval. In *SIGIR2006*, pages 723–724, July 2006.

[38] W. Zhou, V. I. Torvik, and N. R. Smalheiser. Adam: Another database of abbreviations in medline. *Bioinformatics*, 22(22):2813–2818, September 2006.

[39] W. Zhou, C. Yu, N. R. Smalheiser, V. I. Torvik, and H. Jie. Knowledge-intensive conceptual retrieval and passage extraction of biomedical literature. In *SIGIR2007*, pages 655–662, July 2007.