

Semantic Based Retrieval of Biomedical Literature

Raja Loganantharaj and Vardarajan Badri Narayan
Bioinformatics Research Lab, University of Louisiana at Lafayette
logan@cacs.louisiana.edu

Abstract

The published literature in bioinformatics and the biomedical field is rapidly increasing and the total citations maintained by PubMed surpass 16 million. While it is very encouraging to witness the rapid advancement of research in this important area, this enormous volume creates a problem for researchers to keep up with the current literature. Text-based search portals such as Google help to access relevant documents based on search keywords. Unfortunately, the total number of hits for a typical literature search reveals large number of documents.

Most modern search engines help them narrow their search but are not sophisticated enough to understand their needs. Also a user may want to look for documents similar to some other document. Syntax based retrieval systems are incapable of such functionality. The Semantic Web is a vision of next generation World Wide Web where the information is defined with an explicit meaning, which machines can understand, process and integrate without human intervention.

In typical syntactic search, the document is represented in an object space and the search is for the words which define the space. In semantic search, we search for terms in a concept space (a graph of terms occurring in documents linked to each other by the frequency and relationships between them and with which they occur together). Hence by reducing the high dimensional and complex object space into a more manageable automatically-generated, meaningful concept space, customized retrieval and information routing is made easy. In this paper we present an approach that uses semantics to represent and to retrieve relevant documents for a query.

Introduction

Over the last few years, the published materials in the form of journals and conference proceedings have grown exponentially making the web as the most pervasive source of information. No credible research is done in a vacuum; researchers gain inspiration as well as knowledge and information from other's work by reading published literature. When the published

material is over 16 million papers and growing rapidly every day, it becomes a challenging task to find the appropriate papers or reports relevant to one's current investigation. Search portals come to aid to find pertinent documents. For example, we are interested in *breast cancer*; Google search engine brought 125,000,000 entries when we searched with the words "breast cancer". When the search was expanded to *genes responsible breast cancer*, it brought 4,860,000 entries. Pubmed, a search portal for citations having over 16 million documents, was developed by the National Center for Biotechnology Information (NCBI) at the National Library of Medicine (NLM). Pubmed brought 512 citations for the query "genes responsible breast cancer". Semantically the phrase "genes responsible breast cancer" is the same as the phrase "genes causing breast cancer", but Pubmed brought only 123 citations for the query "genes causing breast cancer". When we examined the details of these two search results, the terms *genes* and *breast cancer* were mapped onto MESH-term concepts and the relational terms *responsible* or *causing* did not map onto any MESH concepts. Instead these terms are considered as free text when searching in the documents, which explain the differences in the retrieved number of citations.

Search portals such as Pubmed are said to have a better representation than that of text based search portals since many search words are mapped onto concepts of the MESH terms and the search is conducted in the concept space of the documents. Many text tokens map onto a concept defined by Ontology and such an abstraction reduces the search space and improves the efficiency. Those terms that do not match a token are searched for as plain text.

In this paper we outline a representation, retrieval method and an algorithm based on semantics which improves the quality of the retrieved results. Many text based search engines use vector space model for retrieval. A document is scanned and the stop words that do not convey any meaning or do not have discriminating ability are removed. Root words are then obtained using Porter Stemming algorithm which occur very frequently in a document. The scanned root words excluding the stop words become the bag of words that represent a document. If the same words are

searched, the document associated with those words is brought as the top ranked matched document. This approach implicitly makes a crude assumption that the meaning of a document is solely dependent on the collection of words. In reality, the meaning of the documentation depends only on the arrangement of words in sentences and hence the same bag of words may have different meanings in different documents. This explains lack of precision of the ranking system using bag of words approach.

The paper is organized as following. In section 2, we present related work in this field. It is followed by a brief description of vector space model in section 3. In section 4, we will provide the details of our approach. The paper is completed by a summary and discussion in Section 5.

2. Related Work

A significant amount of work has been done in the area of Semantic Web and semantics-based query retrieval. The Semantic Web project is an architecture woven around the existing web that uses standards, markup languages and other tools to embed semantics. The Semantic Web comprises of the standards and tools of XML, XML Schema, RDF, RDF Schema and OWL. Also there has been a lot of progress in the area of information retrieval from both single and multimodal data sources [1]. The direction of the future is to exploit the virtues of the Semantic Web towards effective Information Retrieval.

Several systems have been designed that use semantics as their underlying architecture. Swoogle [2]: a semantic web search and metadata engine, indexes documents in RDF or OWL format. Swoogle also proposes a prototype for Semantic Web and automates identification of Semantic Web documents. Other systems like such as SHOE [3], Ontobroker [4], WebKB [5], QuizRDF [6] and CREAM [7] use ontologies to annotate and index web documents. Such systems extract information based on the structure of a document and categorize a document into a particular format.

Our approach looks only at the document text and extracts contextual and semantic information from it. Hence it is format independent. Systems like BioPatentMiner[15] use a similar approach. BioPatentMiner searches for concepts that have been patented in patent databases using the BioAnnotator system [16] and then creates a Semantic Web using based on the annotated patents and biomedical dictionaries using RDF and RDFS. SemPub parses the

document text and the free text into UMLS concepts. Examples of systems which map free text into UMLS concepts are IndexFinder[8], SENSE [9], MicroMESH [10], Metaphrase [11], KnowledgeMap [12], PhraseX [13], MetaMap [14]. Most of these systems use NLP techniques or other novel algorithms like phrase subset filtering (IndexFinder) to extract concepts. SemPub uses this as the first step to extract the semantics of text. It goes beyond these systems by building relationships between the concepts. Also it quantifies the importance of each concept and relation using a unique algorithm. The system uses this knowledge to extract contextual information and classify the importance of the text in different contexts. The quantification and ranking of the concept-relation-concept triplets captures the hidden theme of the document.

3. Vector Space Model

Suppose there are K documents in a collection and the total number of independent distinct root words or features in collection of documents are N. A document is viewed as a vector in N-dimensional space. A document $d_i = (w_{1i}, w_{2i}, \dots, w_{ni})$ where w_{ri} is the weight of feature r of document i . The user query is another document and is also represented as a vector in the n dimensional space. The retrieval function computes the similarity between the query vector and the document vectors and produces some predefined number of documents ranked in the decreasing order of the similarity.

Two vectors are said to be similar when one falls onto the other, and they are dissimilar when they are orthogonal. Cosine between a pair of vectors has been successfully used in vector space model to measure the similarity between the user query and a document. The cosine similarity for normalized vectors d_i and d_r is the dot product of these vectors and are given below

$$\text{sim}(d_i \cdot d_r) = d_i \cdot d_r = \sum_{j=1}^{j=n} w_{ji} \cdot w_{jr}$$

Feature weighting or the term weighting influences the effectiveness of the retrieval of the vector space model. The following three factors are considered when computing feature weight: term frequency (tf), document frequency (df), and document length. The term frequency indicates the importance of the term for the content of the document. The document frequency is the number of documents having the term. The importance of a term is inversely proportional to the document frequency, that is, a high value of document frequency indicates low importance of the term in

identifying a given document. When a document is very large, the term frequency will be high and the document may score high, hence the scoring system must be normalized to avoid bias towards a longer document.

Among the various term weighting scheme, term-frequency times inverse-term-frequency and its variation are the popularly used and the simplest form of it defined as:

$$tf \cdot \ln \frac{\text{number_of_documents}}{df}$$

4. Our Approach

Many of the text retrieval systems are based on vector space model that use term frequency and inverse document frequency. Since many terms may map onto a single concept, a term based system may miss relevant documents and hence these systems lack coverage. Suppose a concept, say c_k , is present in all the documents. It is possible that a term denoting the concept c_k may not be in all the documents or different terms denote the same concept in different documents. Hence the inverse document frequency has positive influence in retrieving the document, while in fact the term should not have any influence in the retrieval process since the concept of the term is in all the documents. In summary, a vector space model using terms lacks coverage and focus when retrieving documents.

In our approach, we use concepts instead of terms in representing documents and in their retrieval. By using concepts we can improve the coverage and increase the focus in the retrieval.

The meaning of the document is dependent on the sentences and the structure of the sentences in it. A collection of words and the word frequency alone has only a weak bearing on the semantic meaning of a document. It is possible that two documents having different meaning may have the same bag of words.

This is true even when we represent the documents with concepts, albeit to lesser extent.

We capture the semantics by grabbing the meaning in each sentence. We do this by the reducing the document to a network of concepts connected to each other by relations. To map the words into concept space we use the UMLS Metathesaurus as our Ontology. It is a natural choice since the thesaurus is a superset of ontologies related to the biomedical domain. We also use the UMLS Semantic Network to represent relations between the various concepts. The document is parsed into sentences and each sentence is parsed into phrases. Each word in a phrase is mapped onto concepts and the relations among the words are mapped onto relations between the corresponding concepts. The semantics of a phrase is modeled as a network of concepts and relations among them. A sentence is a sequence of phrases and hence it is modeled as a network of networks.

Let us illustrate our approach with an example drawn from an abstract in PubMed with the following details:

PMID: 16224818

Title: “Macrophage migration inhibitory factor promotes innate immune responses by suppressing glucocorticoid-induced expression of mitogen-activated protein kinase phosphatase-1.”

Consider the following sentence that appears in the abstract of the above document: “The pro-inflammatory cytokine macrophage migration inhibitory factor (MIF) acts as a physiological counter-regulator of the immuno-suppressive effects of glucocorticoids.”

The following six phrases were obtained after pharsing:

P1: pro-inflammatory cytokine macrophage migration inhibitory factor

P2: MIF

P3: acts

P4: physiological counter regulator

P5: immuno suppressive effects

P6: glucocorticoids

The concepts obtained for the terms or combinations of terms in phrases are shown in table 1 as following.

Concept No	Phrase No	Selected word/phrase for the concept	UMLS Concept Number (CUI)	UMLS Semantic Type ID	UMLS semantic type description
C1	P1	pro	C0033382	T123	Biologically Active Substance
C2	P1	Cytokine	C0079189	T129	Immunologic Factor
C3	P1	Macrophage	C0024432	T025	Cell

C4	P1	migration inhibitory factor	C0024429	T129	Immunologic Factor
C5	P2	MIF	C0024429	T129	Immunologic Factor
C6	P2	MIF	C0054504	T129	Immunologic Factor
C7	P2	MIF	C0054505	T129	Immunologic Factor
C8	P2	MIF	C0312359	T126	Enzyme
C9	P2	MIF	C1334507	T028	Gene or Genome
C10	P2	MIF	C1335798	T028	Gene or Genome
C11	P2	MIF	C1366582	T028	Gene or Genome
C12	P3	acts	C0279208	T061	Therapeutic or Preventive Procedure
C13	P4	counter	C0702263	T131	Hazardous or Poisonous Substance
C14	P4	regulator	C0182953	T074	Medical Device
C15	P5	Glucocorticoids	C0017710	T125	Hormone

Table 1: Mapping of terms in phrase into UMLS concepts

There are 54 relations between the 135 semantic types. In the above sentence, the following eight relations were found among the obtained concepts.

- R1: part_of
- R2: location_of
- R3: interacts_with
- R4: produces
- R5: disrupts
- R6: uses
- R7: consists_of
- R8: isa
- NR: No relation

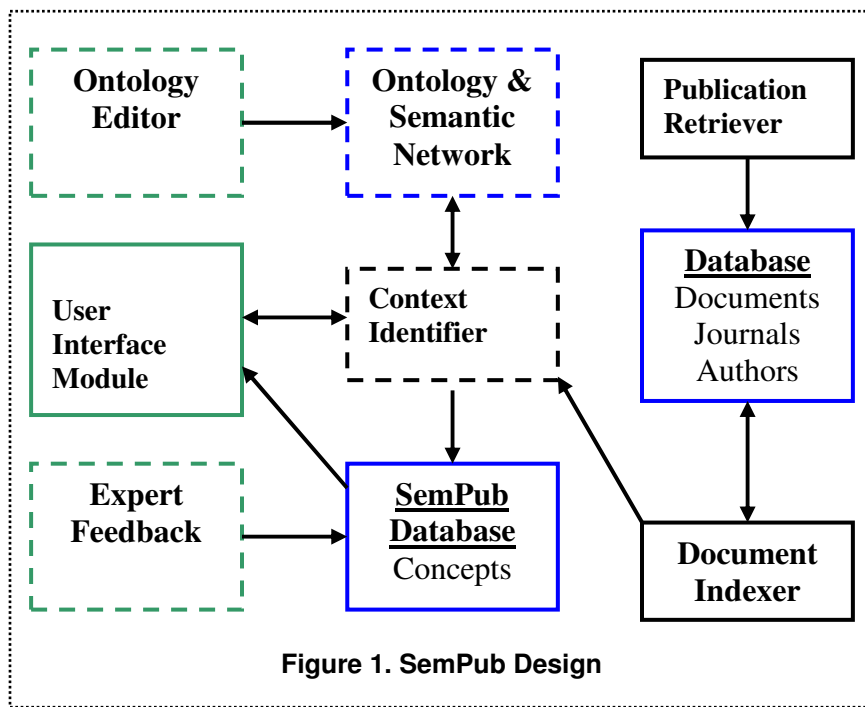
The relations among the 14 concepts are shown by 15X14 table.

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
C1	R5 R8	R4 R8	R4 R8	R4 R8	R4 R8	R4 R8	R3 R8	R4 R5	R4 R5	R4 R5	NR	R3	NR	R3 R8
C2		R4 R8	R3	R3	R3	R3	R3	R4 R5	R4 R5	R4 R5	NR	R3	NR	R3
C3			R4 R5	R4 R5	R4 R5	R4 R5	R4 R5	R1	R1	R1	R2	R5	NR	R4 R5
C4				R3	R3	R3	R3	R4 R5	R4 R5	R4 R5	NR	R3	NR	R3
C5					R3	R3	R3	R4 R5	R4 R5	R4 R5	NR	R3	NR	R3
C6						R3	R3	R4 R5	R4 R5	R4 R5	NR	R3	NR	R3
C7							R3	R4 R5	R4 R5	R4 R5	NR	R3	NR	R3
C8								R7	R7	R7	NR	R5	NR	R4 R5
C9									R7	R7	NR	R5	NR	R4

														R5
C10										R7	NR	R5	NR	R4 R5
C11											NR	R5	NR	R4 R5
C12												NR	R6	NR
C13													NR	R3
C14														NR

Table 2:

The details of the architecture of our semantics-based retrieval system SemPub is provided in [19]. For the completeness of the paper let us briefly describe the architecture of the system. Let us focus on the following three functional modules of SemPub: Publication retriever and preprocessing, Document Indexing, and Query Retrieval.



4.1 Publication Retriever and Preprocessing

The Publication Retriever module is active during this phase of the project. It retrieves the articles from the PubMed database using E-Utils functionality provided by NCBI. The abstracts of the articles are extracted in XML format. For experimental purposes only a specific domain of abstracts are downloaded. (For e.g. all the abstracts that describe a gene family or all

abstracts that talk about breast cancer.) PMC (Pubmed Central) is then used to obtain all the abstracts that are referred to by the abstracts downloaded from PubMed (whenever they are available). The referred abstracts are used by the system to bring context information into the forefront. This module will also retrieve the documents from the database as the new abstracts are released in the public domain. This module brings required information about abstracts and inserts in the local database to be processed by the Document Indexer. In the future it can be coupled with the web

crawler to automatically discover the literature available in the public domain.

The next step in this phase is to parse the XML abstracts and get relevant information about the article. Attributes such as date published, authors and journal information are extracted and stored in a relational database (MYSQL in this case). This step also unearths hidden information such as expert-annotated MESH terms (if present) which describe the article in the best possible way. The Local Database component of the architecture is used to store the information which will be used in the indexing phase. Three major tables Journal, Article and Author are maintained. The Journal table will have the information about all the participating journals along with inferred key concepts addressed in each journal. The keywords are selected high frequency concepts from the published abstracts in that journal. The Abstract Table has the information about the abstract: inferred key concepts, the title words, author list, IDs referred by this abstract and IDs of abstracts referring the abstract. The database approach is preferred over text files to make the data structured and organized.

4.2 Indexing

The Document Indexer does most of the work in this phase. It makes use of NLP and Text processing tools and Ontology and Semantic Network to bring out a concise representation and indexing of the articles for efficient retrieval. With the current implementation there is no Context Identifier, therefore the Document Indexer compares each words against one another to determine the relevancy. Once the Context Identifier is implemented the Document Indexer will select the important words and pass a set of words to the Context miner to determine the context of the set of words. This set of words will determine the subject area of the article. The processing of an article and creating indexes gives the system its underlying semantic nature. It is an iterative process and is done step-wise as follows:

- 1) Each abstract is first searched for its metadata. MESH entries if any are extracted and asterisked terms are differentiated from others. An asterisked MESH term is hand annotated to be one of the most important MESH term described by the document. Also any inherent relationships within the extracted MESH terms are preserved (relations such as descriptor and qualifier). For each term the following steps are performed.

- a) Each term is searched in the UMLS Metathesaurus for a corresponding concept. The concept is the atomic unit of relevance in the system. All articles are described in terms of concepts. (For example, the phrase 'nose bleeding' will map to a concept 'epistaxis')

- b) The concept is then looked for in the semantic network for its semantic type and its abstraction to a higher level is found. (For example, the concept 'Atrial Fibrillation' is of the semantic type 'Pathologic function'.)

- 2) The article is then read and is separated into logical parts. Each abstract is broken into its constituent sentences; the sentences are broken into phrases and the phrases into lexical elements. The lexical elements are the atomic units which describe the semantics of the text. The text is parsed sentence-wise to extract a good semantic representation of the article. A group of words may be closely related but may be used in different sentences in different parts of the text and hence co-relation between them in the given context is pretty low. The words which are close together are more closely related than those far apart. We thus try to map the semantics of a sentence in our system and assume that the meaning of the entire article is cumulative of the different sentences. The steps 1a) and 1b) are repeated for each lexical element, to determine its meaning and its type in our world.

- 3.) An article is thus reduced to a 'bag of concepts' after step 2. We then create an inverted database where concepts are indexed by documents. Hence, given a concept all articles that will mention the concept can be retrieved the database.

- 4.) After we obtain a 'bag of concepts', a semantic tree is built by extracting the relations between concepts from the UMLS semantic network. The terms extracted in step 2 that occur in one sentence are taken two at a time, their semantic types being already known. The relationship between the two semantic types is obtained from the UMLS semantic tree and is recorded in the database as a triplet (the two terms and the relation between them). Thus the article is reduced to a graph; the nodes representing entities or concepts and the edges showing relations between them.

- 5.) The unique feature of SemPub is the scoring mechanism used to score the relevance of a concept, relation or a concept-relation-concept triplet in a given article. This feature separates the SemPub from most modern retrieval system. The scoring is achieved a weighted sum of different factors:

a) The MESH concepts which are hand annotated as important concepts in the article are given more importance than other terms. If a MESH concept is a part of the semantic network, that part is given a higher weight (viz. all the concepts related directly to a MESH concept).

b) The title of the article usually captures the gist of what the article has to say. Hence the concepts and relations occurring in the title are more important than non title terms.

c) Each author has a list of key terms and their relevancy, associated with him, which specify the context or domain he usually writes in. If any of these terms appear in the article, they are scored high to capture that contextual information. A similar strategy is employed for the journal information, though journal terms are scored less because a journal usually has a broader range of topics than an author.

d) The concepts which occur more frequently in the article under consideration and less frequently in its bibliography are assumed to describe a novel idea. These terms are given more weight.

6.) After step 5, we populate the SemPub database, which contains a scored semantic tree for each article. We also build an inverted database as explained in step 4) for fast retrieval of articles. Hence, we use an extension of a vector space model in the system. However we use a semantic matching of terms instead of the conventional Vector Space Model that matches terms syntactically.

The Context Identifier will determine the context of a set of keywords. This module gets the keywords either from the Document Indexer or the User Interface module, and uses the Ontology and Semantic network to determine the context of the set of keywords. It will act on top of the indexer to determine the context of a set of keywords once they are mapped into concepts and relations.

4.3 Retrieval Phase

The retrieval phase is under construction and it will use the User Interface Module for retrieval. The user can enter his query as a set of words, similar to most search engines. The query is then sent to the Document Indexer and like in the indexing phase, it reduces the query to a graph of concepts and relations. All those documents that best describe query graph with a high

relevance score are returned as the best matching documents. The system is also capable of clustering the results in different contexts when the user query is very abstract.

For example, let's assume the user wants to study the effect of **Gene X** in **disease Y**. The system can display the following results:

The documents that are exact matches. It displays all documents that study the effect of Gene X in disease Y and also can display what effect the document talks about. Hence the user can only view those which support one theory, those which say Gene X causes Disease Y.

The system can display all those genes that are closely related to Gene X. A subsequent search can display documents that might describe those closely related genes and their effects on Disease Y.

Those documents that actually describe disease Y can be obtained and thus all the genes that cause that disease can be inferred. Hence the relation between those genes and Gene X can be inferred. Also those documents emphasizing on such relations can be retrieved.

Hence the system would be capable of answering the user query based on the context of requirement. The display system will be capable of letting the user choose the context of search and the area of concentration of the search.

Conclusion

We have outlined our approach for retrieving biomedical literature based on the semantic content. Unlike many search portals that use bag of words, we use concepts in phrases of each sentence and the relations among these concepts to represent documents. By using the concepts and the relations in a document, we increase the coverage and improve the focus and accuracy of retrieval phase. This is a novel approach to represent and retrieve documents with improved accuracy. We believe that our system will retrieve the most relevant documents for a query.

References

- [1] Nawei Chen Technical Report 2006-505 February 2006 *A Survey of Indexing and Retrieval of Multimodal Documents: Text and Images*
- [2] Li Ding et all *Swoogle: A Semantic Web Search and Metadata Engine*

- [3] S. Luke, L. Spector, D. Rager, and J. Hendler. *Ontology-based web agents*. In Proceedings of the First International Conference on Autonomous Agents (Agents97), pages 59{66, 1997.
- [4] S. Decker, M. Erdmann, D. Fensel, and R. Studer. *Ontobroker: Ontology based access to distributed and semi-structured information*. In DS-8, pages 351{369, 1999.
- [5] P. Martin and P. Eklund. *Embedding knowledge in web documents*. In Proceedings of the 8th International World Wide Web Conference (WWW8), pages 324{341, 1999.
- [6] J. Davies, R. Weeks, and U. Krohn. *Quizrdf: search technology for the semantic web*. In WWW2002 workshop on RDF and Semantic Web Applications, 11th International WWW Conference (WWW11), 2002.
- [7] S. Handschuh and S. Staab. *Cream: Creating metadata for the semantic web*. *Comput. Networks*,42(5):579{598, 2003.
- [8] Qinghua Zou, Wesley W. Chu, Craig Morioka, Gregory H. Leazer, and Hooshang Kangarloo *IndexFinder: A Method of Extracting Key Concepts from Clinical Texts for Indexing*
- [9] Yuri L. Ziemann and Howard L. Bleich. *Conceptual Mapping of User's Queries to Medical Subject Headings*. Proc AMIA 1997.
- [10] Elkin PL, Cimino JJ, Lowe HJ, Aronow DB, Payne TH, Pincetl PS and Barnett GO. *Mapping to MESH: The art of trapping MESH equivalence from within narrative text*. Proc 12th SCAMC, 185-190, 1988.
- [11] Tuttle MS, Olson NE, Keck KD, Cole WG, Erlbaum MS, Sherertz DD et al. *Metaphrase: an aid to the clinical conceptualization and formalization of patient problems in healthcare enterprises*. *Methods Inf Med*. 1998 Nov;37(4-5):373-83.
- [12] Joshua C. Denny, Jeffrey D. Smithers, Anderson Spickard, III, Randolph A. Miller. *A New Tool to Identify Key Biomedical Concepts in Text Documents*. Proc AMIA 2002.
- [13] Suresh Srinivasan, Thomas C. Rindfleisch, William T. Hole, Alan R. Aronson, and James G. Mork. *Finding UMLS Metathesaurus Concepts in MEDLINE*. Proc AMIA 2002.
- [14] Alan R. Aronson, *Effective Mapping of Biomedical Text to the UMLS Metathesaurus: The MetaMap Program*. Proc AMIA 2001.
- [15] Sougata Mukherjea, Bhuvan Bamba *BioPatentMiner: An Information Retrieval System for BioMedical Patents* IBM India Research Lab
- [16] L. Subramaniam, S. Mukherjea, P. Kankar, B. Srivastava, V. Batra, P. Kamesam, and R. Kothari. *Information Extraction from Biomedical Literature: Methodology, Evaluation and an Application*. In the Proceedings of the ACM Conference on Information and Knowledge Management, New Orleans, Louisiana, 2003.
- [17] D. Hawking, *Results and challenges in web search evaluation* in Proc. 8th IWWW, Toronto, ON, Canada, 1999.
- [18] V. N. Gudivada and V. V. Raghavan, *Content-based image retrieval systems* IEEE Comput., vol. 28, pp. 18–22, Sept. 1995.
- [19] R. Loganantharaj and Badri N. Vardarajan, *SemPub: An Ontology Based Semantic Literature Retrieval System*", to appear in the proceedings of CBMS 2006.