Query Disambiguation Using Clustering and Concept Based Semantic Web Search For efficient Information Retrieval (QDC-CSWS)

M.Barathi [1] , S.Valli[2]

[1]Department of Computer Science and Engineering, SMK Fomra Institute of Technology, Anna University, Chennai 603103,India
[2]Department of Computer Science and Engineering, Anna University, Chennai 600025,India
bharathi.damu@gmail.com

Abstract: Search queries are short and ambiguous and return a large number of results, in which only a few are relevant to the users. Some search engines suggest a set of related terms to make a user query more specific. Many query expansion techniques are based on keyword and term co-occurrences. These approaches disambiguate the queries and return only a few results that are semantically similar to the user query and ignore the relevant ones. To overcome the ambiguous and short queries, a novel cluster based semantic query expansion technique has been proposed. The proposed work QDC-CSWS first generates the cluster for the initial query results. Secondly, the generated clusters are ranked based on the content similarity to the query. Thirdly, the relevance score is computed against the cluster label and the original query. Fourthly, the ranked clusters are provided as suggestions to the user to disambiguate the query. Finally, the cluster label is enriched by mapping the selected cluster labels with the ontology to extract semantically related concepts. Using only ontology or using only the clustering technique for query expansion might deviate from the query and consequently yield irrelevant results. Sometimes different terms and phrases that co-occur with the initial query are generated by chance and the cluster labels have no semantic meaning. So, to add semantic to these cluster labels, they are mapped on to the ontology concepts, to extract semantically related concepts for expansion. The experimental results show, that this proposed approach has better precision than the existing methods.

[M.Barathi , S.Valli. **Query Disambiguation Using Clustering and Concept Based Semantic Web Search For efficient Information Retrieval (QDC-CSWS).** *Life Sci J* 2013;10(2):147-155].(ISSN:1097-8135). http://www.lifesciencesite.com. 23

Keywords: Query Expansion, Ontology, Clustering, Precision, Word sense Disambiguation, Suffix Tree, Relevance Feedback, Pseudo Relevance Feedback

1. Introduction

Efficient Information Retrieval (IR) systems are required to retrieve the numerous documents on the web. The efficiency of the IR systems depends on the query processing technique, document processing technique, indexing, searching and ranking technique.

The query processing technique retrieves relevant documents with respect to the query given. A single word query may have multiple meanings and would require a disambiguation scenario for obtaining the relevant results. The disambiguation scenario may be a query modification, query refinement or query expansion. The QDC-CSWS uses the query expansion technique to disambiguate the user query. The IR system uses the relevance feedback or pseudo relevance feedback for query expansion [1]. Both the techniques take the top N retrieved documents, and identify the words closely related to the query. The query is enhanced and the enhanced query is fedback to the IR system to increase the number of relevant documents. A relevance feedback system requires user interaction for selecting the relevant documents from the retrieved ones, whereas, the pseudo relevance feedback needs no such interaction. The proposed approach combines both the pseudo relevance feedback and the relevance feedback techniques for the query expansion purpose.

Terms which are closely related to the query can be extracted using lexical resources such as the Wordnet, ontology, Wikipedia, Wiktionary etc. [2], or by using the statistical data available in the documents. Lexical resources provide the traditional synonym or a semantically related word to the query. Therefore, the usage of lexical resources may focus on topics different from the intended query. For example, the query “apple” may be expanded as “fruit”, “food” or “juice” when the lexical resource is used. But the word apple may mean “Macintosh” or “computer”. These interpretations are almost impossible with lexical resources. It may be possible when such resources are domain specific. But, the traditional semantics will be lost. This complexity is overcome in this approach by using domain specific ontology.

The proposed system works as follows.

First, the user enters the initial query, and the top N results are retrieved, preprocessed and clustered using the STC algorithm. Secondly, the clusters are ranked based on the content similarity of the clusters with respect to the original query. Thirdly the relevance score is computed against the cluster label and the original query, Fourthly, the ranked clusters are provided as suggestions to the user to disambiguate the query. Next, to enrich the cluster label, the selected cluster labels are mapped with the ontology to extract semantically related concepts, to focus the search according to the user's needs. Sometimes, different terms and phrases that co-occur with the initial query are generated by chance and the cluster labels have no semantic meaning. So, to add semantic to these cluster labels, these labels are mapped on to the ontology concepts to extract semantically related concepts for expansion. These extracted concepts are augmented with the initial query and fed back to the search engine. This disambiguated query retrieves new and much more focused search results.

The rest of this paper is organized as follows. In section 2 the related works are presented. Section 3 explains the Suffix Tree Document Model and the algorithm. Section 4 describes the main functionalities involved in the QDC-CSWS. Section 5 covers the experimental results. Section 6 is the performance evaluation and Section 7 concludes the work.

2. Related work

Extensive research has been carried out to disambiguate the user query based on query expansion. Erich Schweighofer et al. [3] have proposed a QE technique, which expands a query with the help of a lexical ontology and relevance feedback. Initially, a query is given and the relevance of the results is evaluated by the user. A quantitative term selected from the documents chosen by the user is expanded using ontology, and the semantically related concepts are collected. The over head involved in this work is that, it expects the user to possess domain knowledge about the initial query used in selecting the relevant documents.

Zhiguo Gong et al. [4] used the Term Semantic Network (TSN) which considers co-occurring words during QE. In this case, a query which has multiple meanings retrieves documents pertaining to only one of the meanings.

Agissilaos Andreou [5] used ontology for query expansion. Semantic similarity measures were applied to the concepts extracted for expansion, to induce Word Sense Disambiguation (WSD) through ontology. Boosting factors were used to weigh and rank the concepts extracted from the ontology. The author [10, 32] used lexical resources to identify the context of the query. Neelam Duhan [11] ranked and clustered documents based on relevance. Query reformulation can be used to improve the web search result. The approach [12] learns the user's context according to the user's profile for query reformulation.

WordNet and keyword co-occurrence [19], were used to capture the user's specific context for improving the web search results. Text document clustering improves the performance of the search engines by pre-clustering the entire corpus [7] and post-retrieval document browsing technique [8, 9]. Agglomerative Hierarchical Clustering and K-means clustering are not suitable for post search clustering. Also, these algorithms cannot effectively offer useful topics to group documents. To overcome these shortcomings, Zamir and Etzioni [6] used the Suffix tree clustering algorithm (STC). STC treats a document as a set of phrases and uses them to keep the word semantic order without losing valuable information. An ontology model [13], for personalization was built to improve the web search result. This approach collects and preserves different pieces of information. But, predicting quick user interest change is difficult.

Research in Meta search and distributed retrieval investigates mapping user queries to a set of categories or collection [14]. The authors [15] map a user's interest onto a group of concepts in taxonomy to narrow the search to the user's context. The authors [16] analysed that, due to short and ambiguous queries, a lot of irrelevant results are retrieved. This is because users formulate queries with less than three terms. The approach [36] uses word co-occurrence statistics to resolve ambiguity in the process of query translation.

A number of different existing approaches on query expansion, based on user profile and query logs interpret the semantic meanings of queries and capture user information needs [20, 22, 23, 25, 31, 34, 35]. These approaches collect the user's personal information. The authors [27] use local and global analyses to find the context of the user query. The approach [21] uses the Web as an external data source by issuing queries to collect co-occurrence statistics. These statistics are used to assess the overlap of the contextual entities extracted from the Webpage. The authors [17, 24, 29, 30] improve the web search result by expanding the query based on the cluster results. Fuzzy rules [26] have been used for query expansion. Ontology [28, 33] has been used for query expansion. Semantic similarity measures have been applied to the concepts extracted for Word Sense Disambiguation (WSD) through ontology.

Click through data [37] and a concept-based user profile (CUP) had been used as a concept ontology tree, and the support vector machine (SVM)

had been applied to learn a concept preference vector, for adopting a personalized ranking function for re-ranking the search results. A weak function [39] was used for assessing the similarity between the current query and the knowledge base built from historical users' sessions and the suggestion was generated over an inverted index. Coherency ranking (CR)[39], a domain and database design-independent ranking method for XML keyword queries based on an extension of the concepts of data dependencies and mutual information, was used to retrieve the relevant results.

The QDC-CSWS approach is different from the previous approaches, as it uses the dynamic clustering of search results, a novel cluster ranking based on content similarity to the query, concept extraction from the ontology based on an interactive query suggestion, and disambiguating the query by expansion. Therefore, it makes the user search more focused and improves the web search result.

3. Suffix Tree Document Model and Suffix Tree Cluster Algorithm

The STC clusters the text documents in linear time, based on the common phrases occurring in the text documents. A phrase is an ordered sequence of one or more words.

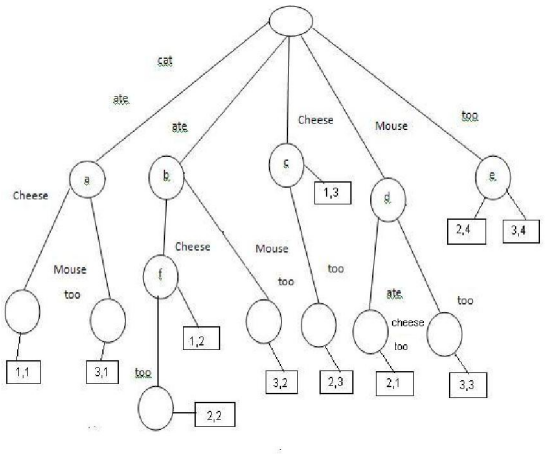
The Suffix tree document model considers a document d with n words. A suffix tree of document d is a compact tree containing all suffixes of document d. The same example [6] is used to describe the suffix tree document. A suffix tree for three documents is given in Figure 1. The node of the suffix tree is represented as circles and each edge represents a phrase. Each internal node has at least two children. Each edge is labeled with a non-empty substring of a document called a phrase, and each suffix node has one or more boxes attached, to indicate where the string originates from. The first number in each box indicates from which document that suffix originated, and the second number represents the position where the suffix starts in that string.

The base clusters are identified by creating an inverted index of documents containing the words present in the phrase. A score S is assigned to each base cluster B using equation (1)

$$S(B)=N_c*N_{wp} \quad (1)$$

In equation (1), S(B) represents the score of the base cluster, N_c represents the number of documents present in the cluster and N_{wp} represents the number of words in the phrase representing the cluster.

Figure 1 An Example of the suffix tree constructed for three documents "cat ate cheese", "mouse ate cheese too" and "cat ate mouse too"

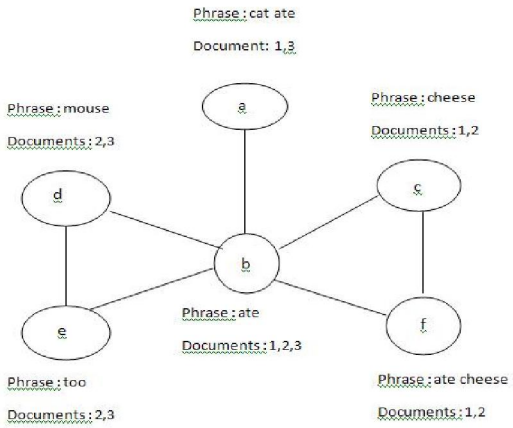


Documents may share more than one phrase. As a result, different base clusters may overlap documents. The base clusters that share more number of documents are merged together to avoid the growth of the base clusters. The similarity [6] between the two base clusters B_i, B_j is calculated using equation (2).

$$\text{Similarity}(B_i,,B_j)= \begin{cases} 1, \text{ iff } (|B_i \cap B_j|/|B_i U B_j|) > 0.5) \\ 0, \quad \text{otherwise} \end{cases} \quad (2)$$

Each node in Figure 1 represents a base cluster. Two nodes are connected if, and only if, the two base clusters have a similarity value of 1, as shown in Figure 2. A cluster is the union of all the documents of all its base clusters. A cluster is a connected component in the base cluster graph. The base clusters a, c, d, e, f are highly connected to base cluster b. So, there is only one cluster in Figure 2. Suppose the word *ate* is considered as a stop word and omitted, then three cluster sets such as {d, e}, {e} and {c, f} are formed.

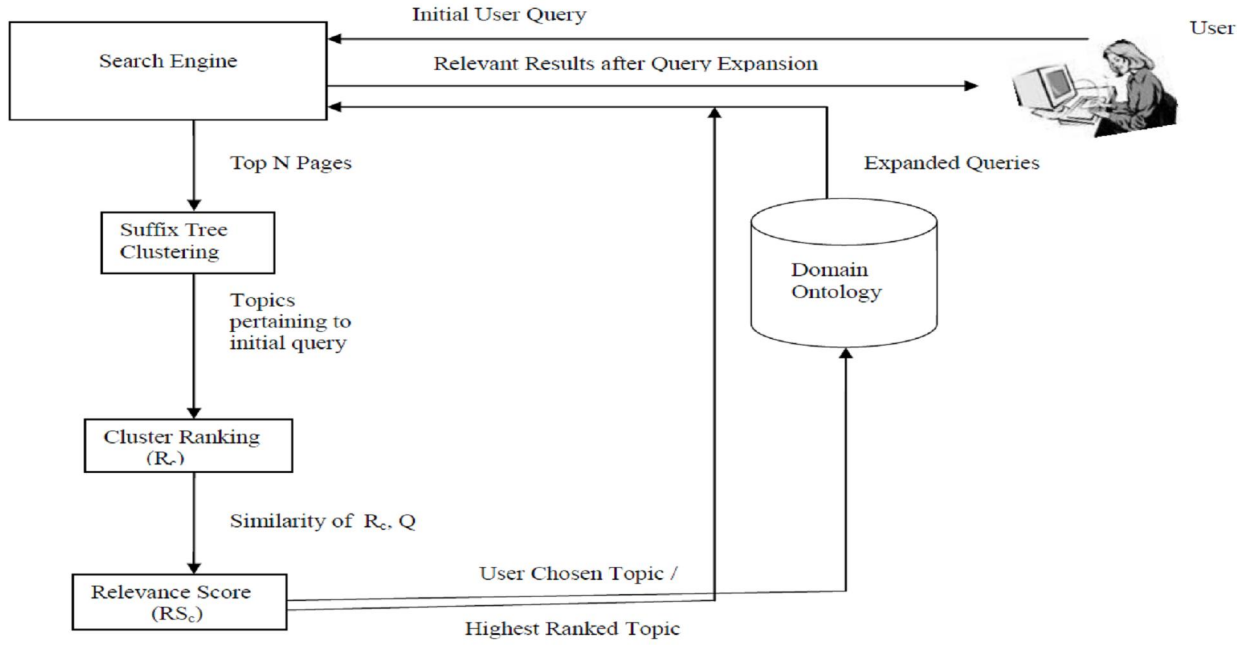
Figure 2 The base cluster graph



4. Query disambiguation and Concept Based Query Expansion

The QDC-CSWS aims at building a focused query expansion (QE) system that analyses a variety of topics from the search result retrieved for the initial query. The QDC-CSWS architecture is shown in Figure 3 and the algorithm is given in Figure 4. Many document clustering algorithms are applied off-line to cluster the document collections. Since the Web search engine's collections are too large, clustering has been done on a smaller set of documents, which is returned in response to a query. The user enters the initial query in the search engine and retrieves the top N pages. All the retrieved N pages are preprocessed and clustered using the STC algorithm. Stemming is a text preprocessing technique, in which the prefix and suffix of each word in the document are removed. The Porter's stemmer algorithm is used to stem the words from each document. Stop words such as articles, pronouns, auxiliaries and prepositions are removed. The initial query results are mainly clustered to disambiguate ambiguous queries.

Figure 3 Architecture of the QDC-CSWS



The main process of QDC-CSWS is

1. To retrieve documents and to generate clusters
2. To perform cluster ranking and Relevance score calculation
3. To generate disambiguated queries suggestions
4. To extract concepts using ontology and expand the query with the extracted concepts

Algorithm Pseudo-code of the QDC-CSWS Algorithm

D ← input documents
{**Step 1**: Preprocessing}
For all d € D do
Apply stemming and remove stop-words in d
Index all the terms using the lucene indexer
Endfor
{**Step 2**: suffix tree clustering}
[Creation of a Generalized Suffix Tree of all sentences]
for each document
{ for each sentence
{if (sentence length > 0)
{insert sentence and all its substrings into generalised suffix tree and update internal nodes with the index to current document while rearranging the tree;
}}}
[Build a list of base clusters]
for each node in the tree
{if (number of documents in node's subtree > 2)
{if(candidateBaseClusterScore>Minimal_Base_Cluster_Score)
{ add a base cluster to the list of base clusters;
}}}
[Merge base clusters]
build a graph where nodes are base clusters and there is a link between node A and B if and only if the number of common documents indexed by A and B is greater than the Merge Threshold;

{**Step 3**: cluster score calculation}
For i= 0 to D do
Calculate $R_c = \sum SR(d_i) /M$
Endfor
Repeat step 4 and 5 until there is no more user suggestion
{**Step 4**: Query Disambiguation based on user suggestion}
Select the user suggested cluster topic
{**Step 5**: Query expansion based on ontology concepts}
Map the user suggested cluster topic with the ontology concepts
Extract top three subclasses and append with the initial query
{**Step 6**: Display user interested relevant results}
Semantically retrieved relevant results are displayed to the user
End

Figure 4 QDC-CSWS algorithm

4.1 Document retrieval and generation of clusters

The first step of the QDC-CSWS process is to retrieve documents from the search engine for the given initial query and to generate the clusters. An initial query is given to the search engine and the top N results are retrieved, processed and indexed using the lucene indexer. The tool, Carrot2, helps in finding the cluster labels for these retrieved documents using the STC algorithm. A cluster may be represented by one or more labels. These labels are the common terms occurring in each cluster. The Suffix Tree Document Model and Suffix Tree Cluster Algorithm are explained in Section 3.

The different clustering labels can be used to identify the context of an ambiguous query. By identifying the right context, we disambiguate the query and get a more specific result. Clustering organizes the different topics returned by the query, and allows the users to have an overview of the query. Users can see the different topics and choose the cluster label that is fit for query expansion. The cluster labels allow users to see the different topics they would have missed, if they had used typical search engines. To get the different cluster labels for query expansion the users need to first enter the initial query. After the initial query results are returned, the web pages are clustered and labelled. The different cluster labels are used to expand the original query.

Unlike WordNet, cluster labels are collection dependent, because the labels are generated by the title and snippets of the different web pages. Cluster labels are terms and phrases that co-occur with the initial query. As the data on the internet changes, the cluster labels also get changed.

As the users browse through the results, they see different terms and phrases and realize that, these are the terms that should be used initially. The terms may have just passed their mind, or maybe they are new, and never thought of or imagined that they can be helpful. So after they find the terms, they modify their original query. When cluster labels are used for query expansion, they can save time because the clusters group the semantically related pages quickly, and get the specific result set instead of browsing through the result of the initial query. The problem with using clustering labels for query expansion terms is that sometimes the different labels are not meaningful. Sometimes different terms and phrases that co-occur with the initial query are generated by chance and the cluster labels have no semantic meaning. So, to add semantic to these cluster labels, they are mapped on to the ontology concepts to extract semantically related concepts for expansion.

4.2 Cluster Ranking

The second step of the QDC-CSWS is to rank all the generated clusters, and to find the relevance score. To rank each cluster, the documents' similarity rank, namely, SRd, is computed using equation (3).

$$SR_d = \frac{N - Pos(d) + 1}{N} \quad (3)$$

where Pos(d) in equation is the position of documents d returned by the search engine, and N is the size of the list. The first ranked document gets a value $SR_d = 1$, while the last one gets a value $SR_d = 1/N$. Then the ranks of the clusters are computed using equation (4)

$$R_c = \frac{\sum_{d \in D_c} SR_d}{M} \quad (4)$$

where R_c is the rank value for each cluster ,D_c are the documents belonging to the cluster, i.e., D_c = {d1,c,. . .,dM,c} and M is the set of documents belonging to each cluster.

Once the documents are clustered and ranked, the relevance score of each cluster is calculated based on the rank of the cluster SR_c with respect to the query as given in equation (5).

$$RS_c = sim(R_c, q) \quad (5)$$

This ranked list of relevance score is presented to the user, for suggestions to disambiguate the query as well as to allow the users to focus their search according to their needs. The cluster labels, documents in each cluster and the cluster score are shown in Table 1.

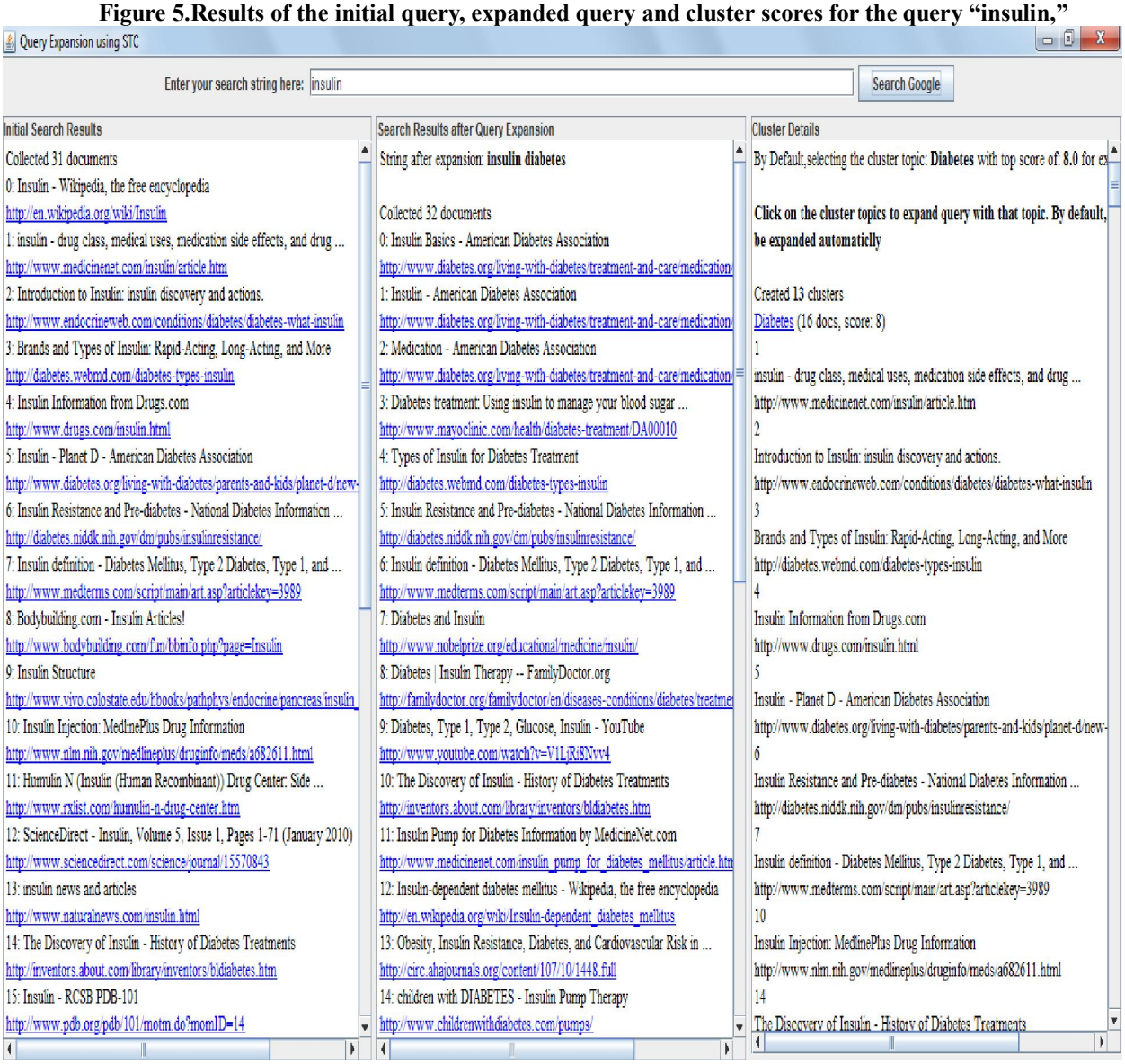
Table 1 Cluster labels and cluster scores for the query "RNA"

Cluster No.	Cluster Labels	Documents in each cluster	Cluster Scores
1	Ribonucleic Acid	0,3,7,8,11,15,27	9.68
2	DNA	2,5,7,8,12,15,20,27	4
3	Information	5,20,23,26,30,31	3
4	Usually Single- stranded	2,18,22	2.65
5	Deoxyribonucleic Acid	7,8,27	2.65
6	Other Topics	1,4,6,9,10,13,14,16,17,19,21,24,25,28,29	0

4.3 Disambiguated query Suggestion

The third step of the QDC-CSWS is to disambiguate the query by user suggestion. The cluster label with the highest relevance score is chosen as the topic that is closely related to the query, and expanded with the initial query to disambiguate the query and to make the user search more focused. Sometimes, the cluster labels which are generated based on different terms and phrases that co-occur with the initial query have no semantic meaning. So, to add semantic to these cluster labels, these labels are mapped on to the ontology concepts to extract semantically related concepts for expansion, which is explained in Section 4.4.

Figure 5.Results of the initial query, expanded query and cluster scores for the query "insulin,"



4.4 Concept Extraction and Query Expansion using Domain Ontology

The fourth step of the QDC-CSWS is to extract the concept based on the user's suggestion. The highest ranked topic or the user chosen topic is mapped on to the domain ontology to extract the semantically related concepts. The Wu & Palmer semantic similarity measure [40] is used to find the semantically related concepts using equation (6).

$$Sim(c1, c2) = \frac{2*depth\ (lcs)}{depth\ (c1)+depth(c2)} \qquad (6)$$

where lcs is the least common super concept of c1 and c2, depth(c1) is the number of nodes on the path from c1 to depth(lcs), depth(c2) is the number of nodes on the path from c2 to depth(lcs) and depth(lcs) is the number of nodes on the path from lcs(c1,c2) to root.

For each cluster, three closely related concepts are extracted from the ontology. These closely related concepts may be the immediate sub classes to the topic, or entities to the classes represented by the topic. These concepts extracted from the domain ontology are chosen as expanded queries. Hence, it can be observed, that the cluster with lexical resources like ontology, helps in adding quality to the expanded queries in terms of semantics. These expanded queries are fed back to the search engine and more relevant documents are retrieved. Figure 5 shows the results of the initial query, the expanded query and clusters with scores.

5. Experimental results

The QDC-CSWS technique was tested on a standard search engine. The tool, Carrot2 interacts with this search engine for the given initial query, and retrieves the top N documents. Several values of N were experimented with, to find the time needed for clustering, ranking and to generate disambiguated queries. The recent version of the Gene ontology, WINE is the domain ontology. The ontology, contains 2,50,671 concepts totally.

The quality of the cluster label depends on the number of terms. A low cluster count would lead to a loss of relevant pages, and a high cluster count would decrease the quality and lead to cluster labels with a single term. The labels with more than one term are more descriptive and specific. So, the maximum cluster count is set to 50. Next, the cluster labels are ranked by the number of documents in each cluster. The cluster label score ranges from 0 to 10. The cluster label with the maximum score is the biggest cluster, and seems to be a more important topic. Then, the cosine similarity is used to compute the relevance score and the threshold is set to 0.5. The cluster merging threshold is set to 0.5. A Low value would merge more clusters together, and lead to group irrelevant web pages. A high value could result in more similar clusters, which lead to duplicate clusters. So the default value is set to 0.5. The last setting is the Phrase Label .This setting assigns weights to multi-word labels against one-word labels. High value gives more number of multi-words and decreases the one word label. To make the cluster label more descriptive and specific, the default value is set from 1.5 to 2, and the maximum number is 10.

6. User Evaluation and performance study

To evaluate the system, a user study of 20 participants is done with the standard search engine. The users are told to use only the search, clustering, query expansion, query expansion with clustering, and ontology functionality of the QDC-CSWS search system. This is done to see which method is faster and to retrieve more relevant pages. Table 2 shows the average time taken by the users to find the relevant web pages from the user study. Processing time includes the time the search engine takes to return the web pages, and the time taken to cluster the web pages. Table 2 shows that the fastest method to find more number of relevant web pages was clustering with ontology concepts. The next quickest method was clustering, the next quickest was query expansion, and the slowest method was the standard search engine.

Table 2 Average time for the four different methods

Search Method	Processing Time
Keyword Search	33.21 s
Query Expansion	22.45 s
Clustering	18.19s
Cluster label + Ontology Concepts	11.27s

The effectiveness of this Information retrieval system is evaluated using precision [7], as given by equation (7). Precision measures the exactness of the search, (i.e.), the percentage accuracy of the retrieved documents.

Precision=Retrieved relevant documents / Retrieved documents (7)

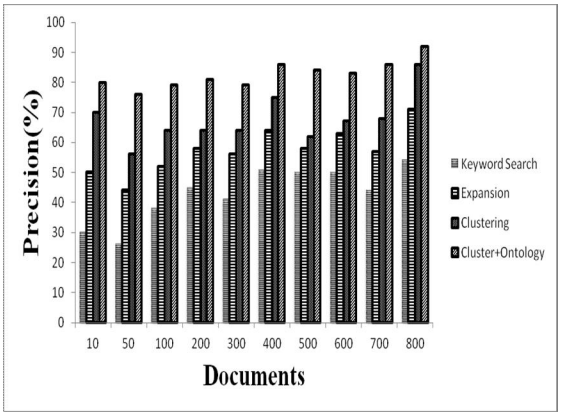
Precision is calculated for several values of N for only the top N documents. The Graphical representation of the change in precision values for the four different methods is shown in Figure 6.

7. Conclusion and Future Work

Query expansion using both the statistical

analysis and lexical resources has been presented in this paper, and it shows the advantages of using the four different methods. The increase in precision after the query expansion, using clustering and ontology shows the efficiency of the QDC-CSWS work. The work of QDC-CSWS can be extended by adding many other statistical analyses, like the latent semantic analysis, and can be clubbed with lexical resources like ontology, Word Net, Frame Net etc. Also, the QDC-CSWS can be made generic by using a generic lexical resource instead of using the domain ontology. The precision graph in Figure 6 shows that there is 10 to 30 % increase in precision for every method. The QDC-CSWS system proves its efficiency by the increase in precision, after the query expansion, using clustering and ontology.

Figure 6 Precision graphs for the four different methods



Corresponding author: M.Barathi ,Department of Computer Science and Engineering, SMK Fomra Institute of Technology, Anna University, Chennai 603103, India . bharathi.damu@gmail.com

References

[1] Aly AA. Using a Query Expansion Technique to Improve Document Retrieval. Int J Inform Techn Know. 2008; 2: 343- 48.

[2] Bernhard D. Query Expansion based on Pseudo Relevance Feedback from Definition Clusters. COLING '10, Proceedings of the 23rd International Conference on Computational Linguistics. 2010; 54-62.

[3] Schweighofer E, Geist A. Legal Query Expansion using Ontologies and RelevanceFeedback. Proceedings of the 2nd Workshop on Legal Ontologies and Artificial Intelligence Techniques LOAIT, 2007; 149-60.

[4] Gong Z, Cheang CW, Hou LU. Web Query Expansion by WordNet. Springer-Verlag Berlin Heidelberg, LNCS 2005; 3588,166-75,

[5] Andreou A. Ontologies and Query expansion. Thesis, Master of Science, School of Informatics, University of Edinburgh, 2005.

[6] Zamir O, Etzioni O. Web Document Clustering: A Feasibility Demonstration. Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, 1998.

[7] Rijsbergen CJV. Information Retrieval. Second Edition, Butterworths, London, 1979.

[8] Allen RB, Obry P, Littman M. AN Interface for navigating clustered document sets returned by queries in Proceeding of the ACM Conference on Organizational Computing Systems, 1993; 166-71.

[9] Voorhees EM. Implementing Agglomerative Hierarchical Clustering Algorithms for use in document retrieval. Int J Inform Proc and Manag. 1986; 22: 465-76.

[10] Barathi M and Valli S. Ontology based query Expansion using word sense Disambiguation", Int J computer sci Inform Sec. 2010; 22-27.

[11] Duhan N, Sharma A.K. A Novel approach for Organizing web Search Results using ranking and Clustering. Int J computer appln. 2010; 5:1-9.

[12] Bouramoul A, Kolladi MK, Doan BL. Presy: A context based query reformulation tool for information retrieval on the web. J Computer Sci. 2010; 6: 470-77.

[13] Golemati M, Katifori A. Creating an ontology for the user profile: Method and applications. Proceedings of the 1st IEEE International Conference on Research Challenges in Information Science, 2007.

[14] Powell AL, French JC, Callan J, Connell M, Viles CL. The impact of database selection on distributed searching. Proceeding of the 23rdAnnual International ACM SIGIR Conference Research and Development in Information Retrieval, (SIGIR'03), ACM, New York, 2003; 232-39.

[15] Ma Z, Pant G, Sheng ORL. Interest-based personalized search. ACM Trans Inform Sys. 2007; 25, 1-38.

[16] Jansen BJ, Booth DL, Sprink A. Pattern of query reformulation during web searching. J Am Soc Inform Sci Tech , 2009; 60: 1358-71.

[17] Bordogna G, Campi A, Psaila G, Ronchi S, "Disambiguated query suggestions and personalized content- similarity and novelty ranking of clustered results to optimize web searches", Int J Inform Process Manag 2012; 48: 419-19.

[18] Weiss D. A Clustering Interface for Search Results in Polish and English. Master Thesis, Poznan University of Technology, 2001.

[19] Barathi M, Valli S. Context Disambiguation

Based Semantic Web Search for Effective Information Retrieval. J Comp Sci. 2011; 7, 548-53.
[20] Tao X, Li Y, Zhong N.A Personalized Ontology Model for Web information Gathering. IEEE Transactions On Knowledge and Data Engineering, 2011; 23, 496-11.
[21] Turan RB, Kalashnikov DV, Mehrotra S. Exploiting Web querying for Web people search. ACM Trans Database Sys. 2012; 37, 7.1-7.41.
[22] Leung KWT, Lee DK. Deriving Concept-Based User Profiles Search Engine Logs. IEEE Trans on Knowledge and Data Engineering, 2010; 22: 969-82.
[23] Leung KWT, Wilfred NG, Lee DL. Personalized Concept-Based Clustering of Search Engine Queries. IEEE Transactions on Knowledge and Data Engineering. 2008; 20:1505-18.
[24] Liu Z, Natarajan S and Chen Y. Query Expansion Based on Clustered Results. Proceedings of the VLDB Endowment.2011; 350-61.
[25] Maskari AA, Sanderson M. The effect of user characteristics on search effectiveness in information retrieval Int J Inform Process Manag. 2011; 47: 719-29.
[26] Chang YC, Chen SM, Liau CJ. A New Query Expansion Method for Document Retrieval Based on the Inference of Fuzzy Rules. J Chinese Inst Engin. 2007; 30: 511-15.
[27] Xu J, Croft WC. Improving the Effectiveness of Information Retrieval with Local Context Analysis. ACM Trans Inform Sys. 2000; 18: 79-112.
[28]Wollersheim D, Rahayu WJ. Ontology Based Query Expansion Framework for Use in Medical Information Systems. Journal of Web Information System. 2005; 1: 1-17.
[29] Kalmanovich IG, Kurland O. Cluster-Based Query Expansion. Proceedings of the ACM SIGIR'09, 2009; 646- 47.
[30] Shen D, Pan R , Sun JT, Pan JJ, Wu K, Yin Jan, Yang Q. Query Enrichment for Web-Query Classification. ACM Trans Inform Sys. 2006; 24: 320-52.
[31] Liu F, Yu C, Meng W. Personalized Web Search for Improving Retrieval Effectiveness", IEEE Transactions on Knowledge and Data Engineering. 2004; 16: 28-40.
[32] Imran H, Sharan A. Thesaurus and Query Expansion. Int J Compu sci Inform Tech. 2009; 1: 89- 97.
[33] Rinaldi AM. An Ontology-Driven Approach for Semantic Information Retrieval on the Web. ACM Trans Internet Tech, 2009; 9: 10.1-10.24.
[34] Chirita PA, Firan CS, Nejdl W. Personalized Query Expansion for the Web. Proceedings of the ACM SIGIR 2007; 7-14.
[35] Li Y, Zhong N. Mining Ontology for Automatically Acquiring Web User Information Needs. IEEE Trans on Knowledge and Data Engineering, 2006; 18: 554-68.
[36] Liu Y, Jin R, Chai JY. A Statistical Framework for Query Translation Disambiguation.ACM Trans Asian Lang Inform Process. 2006; 5: 360-87.
[37] Leung KWT, Lee DL, Wilfred NG, Fung HY. A framework for personalizing web search with concept-based user profiles", ACM Trans Internet Techn. 2012; 11, 17.1-17.29.
[38] Broccolo D, Marcon L, Nardini FM, Perego R, Silvestri F.Generating Suggestions for queries in the long tail with an inverted index. Int J Information Process Manag, 2012; 48, 326-39.
[39] Termehchy A, Winslett M. Using Structural information in XML keyword search. ACM Trans Database Sys 2011; 36: 4.1- 4.39.
[40] Wu Z, Palmer M. Verb semantics and lexical selection in Proceedings of the 32nd Annual Meeting of the Associations for Computational Linguistics, 1994; 133–38.

3/28/2013