Ontology Based Query Reformulation using Rhetorical Relations

Sadaf Inam[1], M. Shoaib[1], Fiaz Majeed[2], M. Inam Sharjeel[3]

[1] Department of Computer Science & Engineering, University of Engineering and Technology Lahore, Pakistan, [2] University of Gujrat, Pakistan, [3] The University of Lahore, Raiwind Road Lahore, Pakistan

Abstract

Web searching is becoming more and more complex due to increased size of information on the web. Users have to face a lot of problems in specifying their needs in the form of query. Query Reformulation techniques are required in order to provide users with the results, according to their expectations. The existing reformulation techniques suffer from the problem of not providing users with expected results in all the cases because the mechanisms used behind those techniques are not much fine and accurate. A technique for query reformulation has been presented in this paper which is based on Cross-document Structure Theory (CST) and Rhetorical Structure Theory (RST). A case study has been carried out to validate the proposed technique. The results are satisfactory as set of reformulated queries generated through this technique is semantically more close to the original query which ultimately provides more relevant data to the user.

***Keywords*: Query Reformulation, Rhetorical Structure Theory, Cross-document Structure Theory**

1. Introduction

With every passing day, internet is being more popular and number of users interacting with it, is getting more and more [15, 17]. With the increased usage of internet, the activity of information searching is also getting more popular and usage of libraries and hard media is automatically reduced. Different Search Engines are playing their role in providing information to users against their queries. Providing the user with his desired results is the main target of the search engines. Different search engines are using different techniques in order to fulfill user needs [7, 8]. Query Reformulation is the one way that helps in providing required information to the users. Interface technologies associated with the search engines also support query reformulation by correcting spellings, presenting alternate terms and some other methods [5, 11].
The question that comes next to mind is why these techniques are required. Basically our current web is getting the form of Semantic web in which we represent our data in machine understandable form and ontologies play an important role in defining semantics of data.

Ontologies provide a way to define a common vocabulary for the purpose of sharing and reuse among different systems [12]. Ontologies help in database interoperability, cross database search, and integration of web services [16], [6]. Now the information on the web is in the form of ontologies and users searching on the web are unaware of the underlying architecture so, the terms they provide for information searching may not be the exact terms presented in the system. But providing them the desired results is the task of web so queries they provide are refined.

Different query reformulation strategies have been used to fulfill user needs. User profiles and his behavior during search have been used for suggesting the alternate terms to the user [3]. Web logs have also been used for reformulating the initial user query [13]. Ontologies have also helped in this matter to reformulate the initial user query and provide him the desired results [9], [19], [18]. Although these techniques help in providing users with the desired results but these might not help in all the cases. Some more reliable strategy is required that could help even the novice users and, that be using correlated documents to get related terms for the user. The strategy that can do this is CST which is based on RST.

Rhetorical Structure Theory [10, 17] is a theory of text organization that has served in many areas from text organization to text generation. RST is the theory that explains the relationships between the text spans of a single document and according to this theory, while creating an extract for a particular answer, a candidate sentence can only be included if something is known about the relation between the candidate sentence and the answer sentence. CST explains the relationships between text spans of different documents which are topically related.
We will be using CST based on RST to present relevant terms to the user. The idea is this, when a user gives a query to the search engine, documents are retrieved against this query and presented to the user. These all documents are topically related. These documents are also submitted to the system that finds if some relationship exists between texts of different document and are those texts are related to the original query [4]. CST helps us in identifying these

relations and what our system does, it extracts those terms, sentences or phrases that hold any CST and RST relationship. These all terms are then collectively presented to the end user along with searched results for helping him to expand his search and get required information [1, 2, 14]. As this method uses the topically related documents to search related terms, this is much better than the previously presented technologies and this also presents new application of CST that it can be used to extract relevant terms based on CST relations.

2. Materials and Methods

The architecture of proposed technique is demonstrated in Fig. 1.

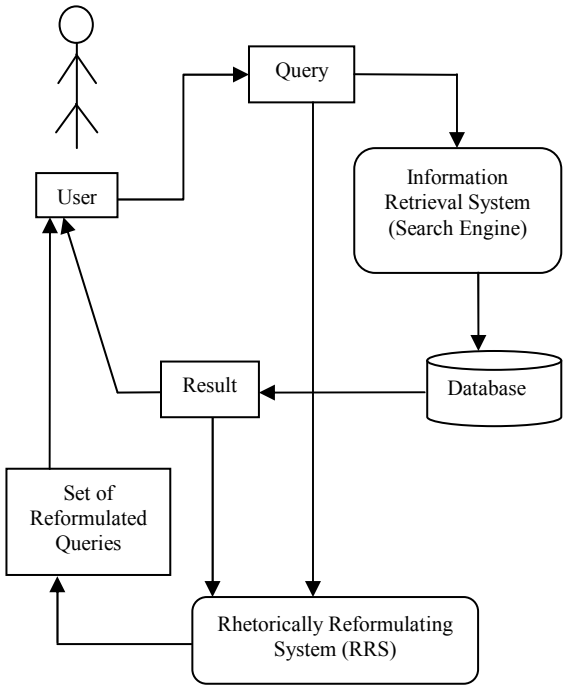

Fig. 1: Rhetorically Reformulating System

Now here is brief explanation of each component of the presented system.

Query: This is the text that user gives to the Search Engine in order to get his required information.

Information Retrieval System: The system that receives the query from user and finds the information for user from the database. Search Engines can be Google, Yahoo and any other.

Database: This is the era of semantic web and as far as Database component is considered, we assume that information is distributed on different peers and we are also assuming that all knowledgebase is in the form of ontologies. Ontologies are concept hierarchies that help in intelligently answering a query. Ontologies are the means that remove the semantic gap between user's view and Database Designer's view. So we assume all database information within a domain is specified through ontologies.

Results: These are the results that are generated for the user against his request from the underlying database in which information, we assume, is in the form of ontologies.

Set of Reformulated Queries: is the expanded set of queries that is generated from our system for user so that he/she may research in order to get his desired results from the system.

This Rhetorically Reformulating System (RRS) is explained in the Fig. 2. It receives the initial user query as input as well as it gets retrieved results from database. Rhetorical Relations are predefined in it. While implementing, we will be taking the assistance of Support Vector Machine (SVM) in our system as its component in order to identify relations between two text spans. What change we need to consider or assume in its (SVMs) implementation, is that we will be defining our own set of relations that, we consider, can exist between different text spans. Now RRS checks two text spans (that in our case are phrases or words) and if some relation (from pre-defined relations) exhibit between these, these terms or phrases are added to set of reformulated queries which was initially empty.

3. Proposed Algorithm

3.1 The Algorithm

Building a module that could assist search engine in reformulating the user given query using rhetorical structures and assuming that information at peers is in the form of ontologies. Algorithm is presented in Fig. 3.

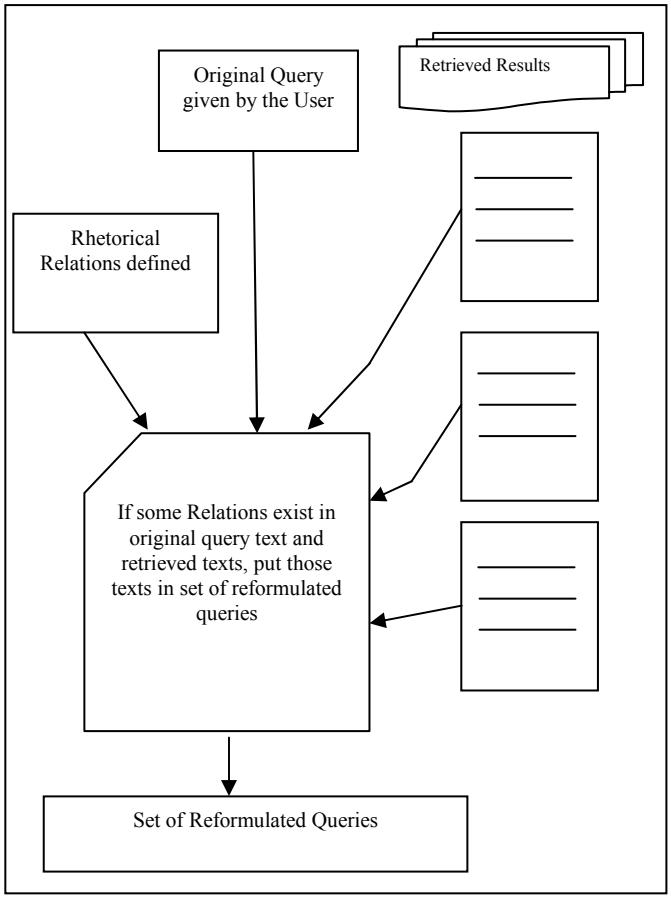

Fig. 2: Rhetorically Reformulating System

Now according to our algorithm, first of all we assume that our reformulated query set is empty. We are using Google as Search Engine and we put query "Teaching jobs" that is taken as original query text.

Now we further consider that an array of pointers contains the addresses of this original query text and retrieved results. We also have a set of pre-defined relations that have been defined for cross-document structures. Now for all retrieved documents, we check that whether any relation exists between text of the query and retrieved results or not. Table 1 below shows the relationships between different texts of original query and retrieved results.

Input:

- OQ: Original Query
- R: Set of Pre-defined Relations defined for any two Text Spans
- RR: Retrieved Results (Set of Documents, retrieved against original query from database)

Output:

- Q': Set of Rhetorically Reformulated Queries (RRQ)

Algorithm:

- **INITIALIZE** Q' to NULL
- **SUBMIT** the user Query to Search Engine.
- **RETRIEVE** the results from Information Retrieval System and consider them as text1, text2, and so on text10, assume we will be considering first 10 retrieved documents.
- **DISPLAY** these results to the user.
- **STORE** OQ (Original Query) and RR (Retrieved Results) in an array of pointers, each pointer pointing to the texts retrieved.
- **DO**
  - CHECK whether any relationship from R (Pre-defined Relations) exists between two text spans or not
  - **IF** (*Some relation exists from pre-defined relations R*)
    - Combine the texts having relation with the original list of reformulated queries that is

      Q' = Q' U {terms/ texts having relations}
  - **ENDIF**
- **WHILE** (Retrieved Results are present in the record)
- **PRESENT/ DISPLAY** the Q' to the user

Fig. 3: Algorithm for representing the system

3.2 Original Query

"Teaching jobs"

Format of the Table1and relations described in it has been taken from [20] and data in the table is based on the results, provided by the Google against original query.

Based on the documents retrieved by Google, we check the relevancy of terms by seeing if any cross document relationship exists between documents. If relationships exist between any two text spans, terms are extracted from these text spans and are added to set of reformulated queries.

Table 1: Query Result

Relationship	Description
Identity	Same text appears in more than one locations
Equivalence (Paraphrase)	Two text spans have the same information content
Subsumption	S1 contains all information in S2, plus additional information not in S2
Contradiction	Contradiction Conflicting information
Citation	S2 explicitly cites document S1
Elaboration	S1 elaborates or provides details of some information given more generally in S2
Summary	S1 summarizes S2.
Reader Profile	S1 and S2 provide similar information written for a different audience.

We used Google to validate the proposed technique and results are being verified. We provided original query in the search engine, results were retrieved from the database as usual and these results were submitted to the proposed system that checks whether any rhetorical relation exists between text spans of retrieved documents. These documents are different but all are topically same as have been retrieved against a single query. We check the presence of any relation between texts. If some relation exists between two different spans of texts, these text, terms or phrases have been added to the set of reformulated queries that is finally presented to the user along with original retrieved results. These terms helped to expand the user's search.

After having results and original query, some relation were identified. Depending on these relations, the specific text was chosen and put in the set of reformulated queries. Set of Reformulated queries after running algorithm as follows:

Q' = { Teaching jobs in Pakistan,
Education jobs,
Online Teaching jobs,
Private jobs,
Government jobs,
Career Opportunities,
Employment opportunities,
Teaching Vacancies,
Teaching specialist jobs in the UK,
College Teaching and Learning
}

And finally set is presented to the user. This process is similar to Keyword Analysis which is a process of Search Engine Optimization.

4. Analysis of proposed Technique with another example

Recall and precision decides the performance of an information retrieval system. For the purpose of evaluation of our system, we will be considering that we have 100 documents in our database. Now according to our algorithm, we will run a query on our proposed system; all the steps that we require for evaluation of the system are presented below.

- Submission of initial query Q_i to the system
- Retrieval of related document against the Q_i
- Submission of Retrieved results plus Q_i to proposed system
- Extraction of terms against the relations that we say, are reformulated query terms

This is what our system is doing, and now what we will be doing additionally for evaluation of our system is

- Submission of queries from newly generated set to the search engine
- Analysis of the results that are retrieved against these new queries
- Analyzing precision and recall of the system

This is the whole process that we have to go through. As case study, we got a collection of 100 documents for the evaluation of our proposed approach. We consider that we have 100 files in our collection that can be in different formats whether pdf, ppt or these can be text files as well and some of these will be relevant to our query and others not, whereas some documents will be partially relevant.
Table 2 and Fig. 4 shows the ratio of documents with respect to relevancy in which total relevant are 77 and 23 are irrelevant.

Table 2: Degree of relevance of Documents

Results	Percentage
Most Relevant	46
Average Relevant	17
Less Relevant	14
Not Relevant	23
Total Documents	100

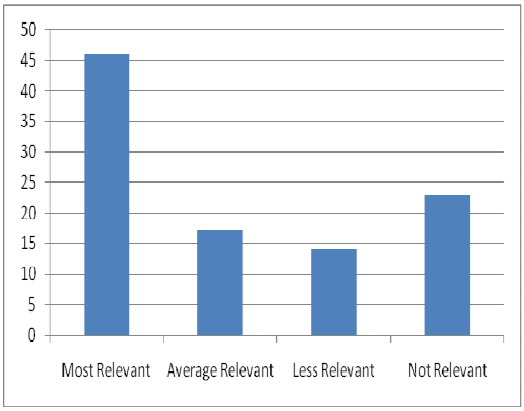

Fig. 4: Graphical Representation of documents relevance

Above is the ratio of all 100 documents. When we gave query to the system, 60 documents were retrieved and we analyzed that 41 documents are relevant and remaining 19 are irrelevant documents. Now we calculate Precision and Recall before Query Reformulation.

4.1 Precision and Recall before Query Reformulation

Total Documents: 100

Relevant: 77

Non Relevant: 23

Query: Carbon Consulting

Retrieved Documents: 60

Relevant out of Retrieved: 41

Precision:

P = |Relevant AND Retrieved| / |Retrieved |

P = 41 / 60 = 0.68

Recall:

R = |Relevant AND Retrieved| / |Relevant |

R = 41 / 77 = 0.53

So this shows that total 53% Relevant Results are retrieved and 68% of retrieved results are relevant.

4.2 Building Queries against our proposed Technique

Now we take the query and review the results accordingly. We suppose the query "Carbon Consulting" and we consider that we have 100 documents in our corpus that we will be working upon. Out of these 100, we further take the retrieved documents only from which we will be analyzing one relevant document D1 for finding relations and thus generating query terms, whereas other documents, we will be using for comparison of initial queries and generated queries by calculating precision and recall of the retrieved documents (Fig. 5 and Table 3).

Initial Query: Carbon Consulting

Analysis of Relations between Texts:

Text1: is our query. i.e; Carbon Consulting

Presentation of Results for the document D1:

Carbon Asset Development And Management, Carbon Management Consulting has the expertise to do both. We understand energy and environmental policy, as well as energy and environmental markets. By interpreting and anticipating policy developments, **CMC** aims to help various stakeholders benefit from the multiple advantages the **Clean Development Mechanism (CDM)** presents. **CMC** promotes projects in various geographic areas notably **India, China, South East Asia and Latin America**.

Fig. 5: Retrieved Contents of the Document D1

Table 3: Terms generated against Relation

Relation	Terms
Identity	Carbon Consulting
Subsumption	Carbon Management Consulting
Summary	CMC
Reader Profile	Carbon Projects for India, china

Now we use new queries that have been generated using Rhetorical Relations. These all queries are more specific and give us more accurate results as compared to previous results. For example, here we will be showing results of two rhetorically generated queries.

New Query 1: Carbon Consulting Services

Now we again calculate precision and Recall and see how it is working now. When we give this query to the system, this gives us the following results.

4.3 Precision and Recall after Query Reformulation

Total Documents: 100

Relevant: 77

Non Relevant: 23

Query: Carbon Consulting Services

Retrieved Documents: 68

Relevant out of Retrieved: 59

Precision:

P = |Relevant AND Retrieved| / |Retrieved |
P = 59 / 68 = 0.867

Recall:

R = |Relevant AND Retrieved| / |Relevant |
R = 59 / 77 = 0.766

So this shows that total 76% Relevant Results are retrieved and 86% of retrieved results are relevant. This shows that reformulated query returns more relevant results that could satisfy web searcher. Fig. 6 depicts the graph representation for above results whereas Table 4 gives precision and recall values.

Table 4: Precision and Recall Before and After Query Reformulation

	Precision	Recall
Before	0.68	0.53
After	0.86	0.766

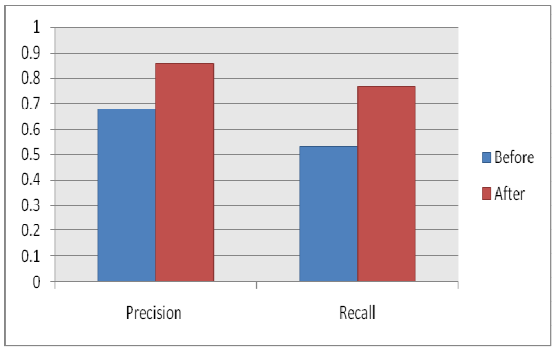

Fig. 6: Precision and Recall before and after Query Reformulation

4.4 Results of Queries before Reformulation

Then we considered five queries and analyzed results against those queries. Table 5 shows our results.

Table 5: Precision and Recall of Initial Queries

	Terms	Retrieved Documents	Relevant Documents	Precision	Recall
Q1	Carbon Consulting	60	41	0.68	0.53
Q2	Carbon Reduction	57	35	0.61	0.45
Q3	Consulting Group	70	47	0.67	0.61
Q4	Emission Reduction	50	27	0.54	0.35
Q5	Consulting Company	45	25	0.55	0.32

Fig. 7 demonstrates graphically precision and recall of initial queries.

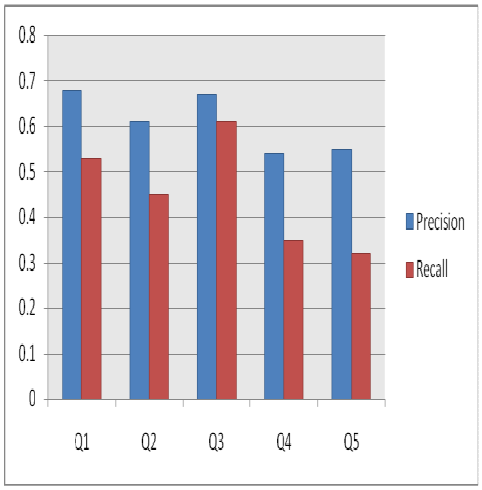

Fig. 7: Precision and Recall of Initial Queries

4.5 Results of Queries after Reformulation (with Reformulated Terms)

Table 7 shows results of reformulated queries.

Table 6: Precision and Recall of Reformulated Queries

	Terms	Retrieved Documents	Relevant Documents	Precision	Recall
Q1	Carbon Consulting Services	68	59	0.86	0.76
Q2	Carbon Emission Reduction	60	55	0.91	0.71
Q3	Carbon Inventory Management	59	45	0.76	0.58
Q4	Carbon Consulting Team	70	64	0.91	0.83
Q5	Carbon Consulting Services	80	72	0.90	0.93

Graphical representation for above results is shown (Fig.8).

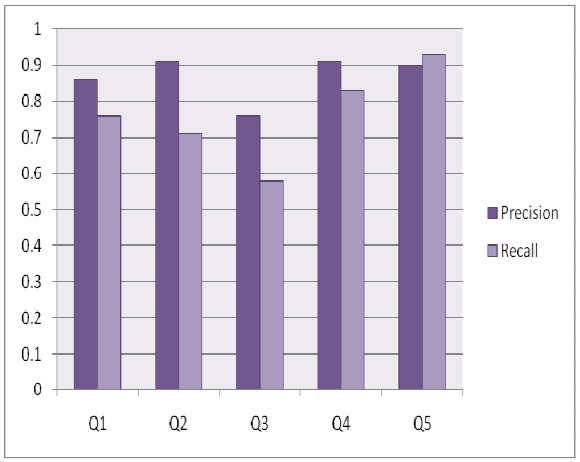

Figure 8: Precision and Recall of Reformulated Queries

Combined results for comparison are shown below (Table 7 and Fig. 9):

Table 7: Precision and Recall before and after Reformulation

	Before		After	
	Precision	**Recall**	**Precision**	**Recall**
Q1	0.68	0.53	0.86	0.76
Q2	0.61	0.45	0.91	0.71
Q3	0.67	0.61	0.76	0.58
Q4	0.54	0.35	0.91	0.83
Q5	0.55	0.32	0.9	0.93

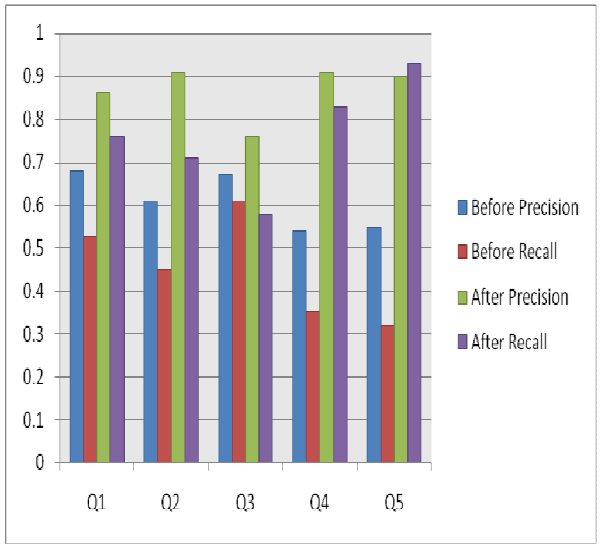

Figure 9: Precision and Recall before and after Reformulation

The results of precision and recall shows that after reformulation, we have more better results that shows that the queries or terms generated after rhetorical reformulation provides web searchers with more satisfactory results.

Comparison of Technologies

Query Reformulation has got much attention in past and many techniques have been presented for reformulating user query in order to provide him with best and relevant results. While reformulating user query, it has been assumed in many cases that underlying information structure is specified through ontologies as ontologies provide a better way of information representation in semantic web. Among different technologies, ontologies themselves have been playing a vital role for reformulation. And as far as RST is concerned, this has also been used in text generation and organization. This highlights the relationships among different spans of a text document. After that CST is presented which describes the relationships between texts of different document that are topically same. So, I got an idea that this can be used for Query Reformulation. Idea is basically that in which terms or sentences, we find relations; we extract those sentences or terms from the text and combine all this text in a set of reformulated queries.

The proposed technique differs from the previous ones in the sense that all previous techniques have been using ontological information or profile in order to reformulate user query that gives you alternate terms but that all may not be valid in all cases. Whereas the proposed technique is using a different theory to generate set of reformulated queries. This difference makes it more interesting and this approach will provide more expected and desired results.

5. Conclusions

A system has been designed for providing a set of reformulated queries to the user against his/her initial query in order to provide the best possible results that match user's needs. The idea is based on the relations that exist between text spans of topically same but different documents. The technique has been proved using Google as search engine. The experiments show that this strategy works in the better way as compared to previous techniques and the set of generated reformulated queries contains the terms that are semantically relevant to initial query.

REFERENCES

[1] Akahani J. I., Hiramatsu K., and Satoh T., "Approximate Query Reformulation for Ontology Integration", 2003.
[2] Akahani J. I., Hiramatsu K., and Satoh T., "Approximate Query Reformulation based on Hierarchical Ontology Mapping", 2003.
[3] Asfari O., Doan B. L., Bourda Y, Sansonnet J. P., "Improving User Query Processing Based on User Profile and Task Context ", 2010
[4] Asfari O., Doan B. L., Bourda Y, Sansonnet J. P., "Personalized Access to Information by Query Reformulation Based on the State of the Current Task and User Profile", 2009 Third International Conference on Advances in Semantic Processing, Malta, 2009
[5] Asfari O., Doan B. L., Bourda Y., Sansonnet J. P., "A Context-based Model for Web Query Reformulation", International Conference on Knowledge Discovery and Information Retrieval" (KDIR 2010), 25-28 oct 2010, Valencia, Spain
[6] B. Chandrasekaran and John R. Josephson, Ohio State University V. Richard Benjamins, "What Are Ontologies, and Why Do We Need Them?", University of Amsterdam,
[7] Bouramoul A., Kholladi M. K., Doan B. L., "PRESY: A Context Based Query Reformulation Tool for Information Retrieval on the Web", Journal of Computer Science 6 (4): 470-477, 2010
[8] Calvanese D., Giacomo D. G., Lembo D., Lenzerini M., Rosati R., "Query Reformulation over Ontology based Peers", 2004.
[9] Calvanese D., Giacomo D. G., Lembo D., Lenzerini M., Rosati R., "What to Ask to a Peer: Ontology-based Query Reformulation", 2004.
[10] Forsbom E., "Rhetorical Structure Theory in Natural Language Generation", 2005.
[11] From the book "Search User Interfaces", published by Cambridge University Press. Copyright © 2009 by Marti A. Hearst.
[12] Gruber, Thomas R. "A translation approach to portable ontology specifications", Knowledge Acquisition 5, June 1993.
[13] Huang J. and Efthimiadis E. N., "*Analyzing and Evaluating Query Reformulation Strategies in Web Search Logs*, 2009.
[14] Jansen B. J., Zhang M., Spink A., "Patterns and Transitions of Query Reformulation during Web Searching"
[15] L. Miller, Ontologies and Metadata, *A Draft Discussion of issues raised by the Semantic Web Technologies Workshop, 22-23 November 2000*. http://ilrt.org/discovery/2000/11/lux/
[16] Ling Liu and M. Tamer Özsu (Eds.), "Ontology" to appear in the Encyclopedia of Database Systems, Springer-Verlag, 2008.
[17] Mann B., "An Introduction to Rhetorical Structure Theory (RST)", August 1999.
[18] Munir k., Odeh M., McClatchey R., "Ontology Assisted Query Reformulation using the Semantic and Assertion Capabilities of OWL-DL Ontologies", 2008.
[19] Necib C. B. and Freytag J. C., "Using Ontologies for Database Query Reformulation", 2004.
[20] Zhang Z., Goldensohn, S. B., Radev, D. R., "Towards CST-Enhanced Summarization", 2002.

Sadaf Inam received MS degree from University of Engineering and Technology (UET) Lahore Pakistan in 2012. Her research interests include Information Retrieval. She has published 1 paper in refereed journal in the above areas.

M. Shoaib received PhD degree from University of Engineering and Technology Lahore Pakistan in 2006. He is currently associate professor in this university. His research interests include information retrieval and data mining. He has published more than 30 papers in refereed journals and international conference proceedings in the above areas.

Fiaz Majeed received MS degree from COMSATS Institute of Information Technology (CIIT) Lahore Pakistan in 2009. He is currently PhD scholar in University of Engineering and Technology Lahore Pakistan. His research interests include Data Warehousing, Data Mining, Data Streams and Information Retrieval. He has published 7 papers in refereed journals and international conference proceedings in the above areas.

M. Inam Sharjeel received his M.Sc. degree from University of Agriculture, Faisalabad. His research interests include Information Extraction and presentation of Information for web users.