Constraint determination algorithm for query reformulation for personalized educational relational databases

Ebru Özpolat and **Gözde Bozdağı Akar**

Dept. Electrical and Electronics Eng., Middle East Technical University, 06531, Ankara, Turkey

There is a huge number of existing heterogeneous information sources which could be used in an educational context. The standard approach to integrating these sources is to build a global schema that relates all of the information in the different sources, and to pose queries directly against it. The problem is that schema integration is usually difficult. Alternatively, the user queries can be processed in a hidden manner from the user and the requested results can be returned. In this paper, we address this problem: How to identify the sources of information that are required to answer a query and determine how the data from them should be combined to give the most precise results the user requested. In addition, we propose a personalized approach to the stated problem such that different users may be returned different results to the same query based on their preferences.

Keywords e-learning; educational; personalization; query reformulation; constraint determination

1. Introduction

With the developments in Internet access and information technologies, the state of the art in education has changed tremendously in the recent years [1,2]. Now web-based learning, e-learning, on- line-learning, assessment, training plays an important role in our educational system. This gives rise to emergence of e-learning systems[3]. The fundamental particle of next-generation e-learning systems is the learning objects. Learning objects represent a major paradigm shift away from the course concept: traditional unit of learning. The problem with the "course" is that it is not very flexible and it is difficult to re-purpose. Learning objects stored in a database and properly tagged for easy searches, are designed specifically for flexibility and re-use. In parallel many standard organizations are working on a model for description of a learning object with the aim of facilitation of search, evaluation and exchange of products, components and learning content such as IEEE LOM[4] and SCORM[5].

As a result of the advances in e-learning systems, schools and publishers prefer to present their learning content as learning object repositories. Due to this tendency towards learning objects, there is a huge number of existing heterogeneous information resources (learning object repositories) which could be used in an educational context. Learners and/or teachers should be able to benefit all of these resources, whenever they pose a metadata query to get the necessary information regarding their research.

The standard approach to integrating these sources is to build a global schema that relates all of the information in the different sources, and to pose queries directly against it [6]. The problem is that schema integration is usually difficult. Combining local schemas into a global schema requires central administration, which brings the limitation to local data resources not to change frequently or seriously, since such a change may break up the defined mapping rules between the local schema and the global schema [7,12, 13]. Alternatively, the user queries can be processed in a hidden manner from the user and the requested results can be returned. In this paper, we address this problem: How to identify the sources of information that are required to answer a query and determine how the data from them should be combined to give the most precise results the user requested. In addition, we propose a personalized approach to the stated problem such that different users may be returned different results to the same query based on their preferences. In other words, we propose an algorithm for constraint determination based on the user's preferences. In order to get the selected result set for the given generic input query in a hidden manner, we reformulate the input queries according to predefined mapping rules. These mapping rules are composed of user specific constraints. Although the issue of determining user-specific constraints to be used as mapping rules is not directly addressed by any study in the literature, we developed our algorithm in the light of some closely related works in the literature, which provide us different point of views. The closely related works are query mapping across heterogeneous information resources by translating query constraints [8], dynamically constraint mapping across web interfaces [9] and COMA, which is a system for flexible combination of schema matching approaches [10]. In the following section, we give the implementation procedure of our algorithm. Also, special cases considered when implementing this algorithm are explained. Experimental results based on [11] are given. We conclude with observations and possible future work.

2. Constraint Determination Algorithm

In order to get the selected result set for the given generic input query in a hidden manner, we reformulate the input queries according to predefined mapping rules. These mapping rules are composed of user specific constraints.

In order to determine these constraints, the following algorithm is developed:

Input: *C:*User's initial query's constraint specified in where clause

Q(C): Original query with the initial constraint

Output: *R(C):* The specified constraints to be used as a mapping rule in query reformulation.

Procedure:

1) Get all the results; i.e. metadata values of all attributes, for C in a table format as shown on Table 1, where columns are attribute names (A's) and rows are corresponding predefined metadata values(V(A)'s).

Table 1 Format of the result set for Q(C)

A_1	A_2	...	A_j	...
$V(A_{11})$	$V(A_{21})$	…	$V(A_{j1})$	…
$V(A_{12})$	$V(A_{22})$	…	$V(A_{j2})$	…
...	…	…	...	…
$V(A_{1i})$	$V(A_{2i})$	…	$V(A_{ji})$	…
...	…	…	...	…

2) Let the user select the desired result subset from the whole result set.

3) The rest of the procedure uses this selected result subset to determine the specified constraint for the given input query constraint.

We investigated the possible patterns seen on the selected result set in three main cases. The first case covers the group, where an attribute, takes the same predetermined value for all rows.

The second case covers the group, where an attribute does not take only any one of its predetermined values for any rows.

The third case concentrates on the attributes, having various predetermined values in different number of occurrences and different orders. In other words, this group covers the columns with irregular patterns. The details of this case are explained in detail in section 2.1.

In figure 1, a flowchart of the procedure covering all of these groups in a general manner is shown.

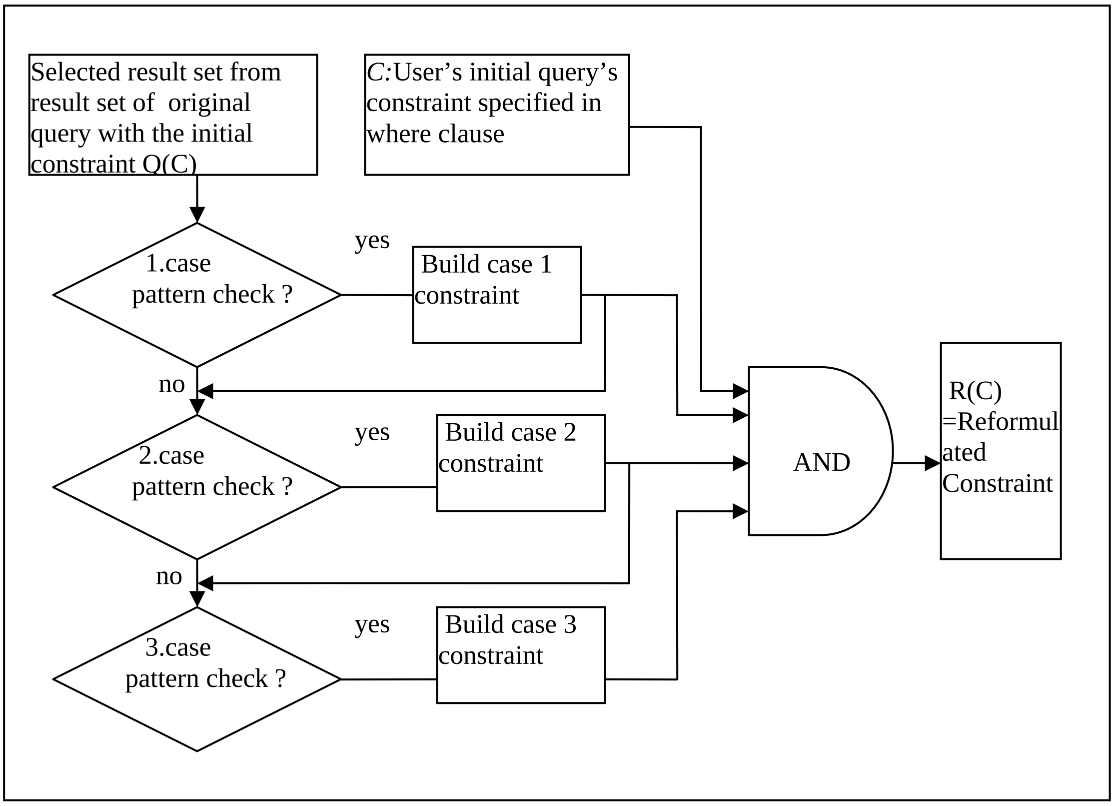

Fig. 1 Flowchart of the Procedure

2.1. Special Cases Covered by the Constraint Determination Algorithm

Special cases occur for the columns with irregular patterns. Before going into details, we can separate irregular patterns; i.e. third case mentioned above, into two basic categories, namely case 3-A and case 3-B.

Case 3-A covers the following patterns as shown on the following table-1: For some rows; if a column has the same one of its predetermined values, while another column(s) has/have the same one of its corresponding predetermined values for the same rows.

On the other hand, case 3-B covers the following patterns illustrated on the table-2: For some rows; if a column has the same one of its predetermined values, while another column(s) has/have the same one of its corresponding predetermined values for the rest of the rows.

Moreover, there are some special cases regarding the irregular patterns. These special cases are taken into consideration, when implementing our algorithm in order to improve the efficiency of the algorithm. Giving attention to these special cases provide us the ability to avoid repeating any covered constraint or not to forget any uncovered pattern with the searched characteristic.

Regarding the special cases, one of them is that when case 3-A covers all the rows for any attribute, then there is no need for searching case 3-B, since in such a situation, case 3-A encapsulates case 3-B.

Another one is for patterns in case 3-A category, such that it should be prevented that for any attribute any found constraint should not be a sub element of other attribute's found constraint for the same selected rows.

There is one more special case for patterns in case 3-A category. That is, for case 3-A category, the following case should be covered: While an attribute takes the same one of its predetermined values for N rows, another attribute takes the same one of its predetermined values for N/x rows and another attribute takes the same one of its predetermined values for N/y rows. Moreover, these N/x and N/y rows list of each attribute can have intersection in between.

Another case is that, if case 3-A does not cover all the rows for any attribute,i.e. if there exist still uncovered rows , then searching for case 3-B is definitely required. In this case, a bridge constraint may be required between case 3-A and case 3-B.

The existence of a bridge constraint causes the emergence of a new special case. If there is a bridge constraint, it should be prevented that any part of bridge constraint should not complement with any part of connecting case 3-B constraint.

3. Evaluation of the Algorithm

We performed a comprehensive evaluation of our constraint determination algorithm on several relational databases covering educational context. For our evaluation, we used educational databases such that their column names correspond to metadata names and their rows correspond to the related metadata's values, which are predefined. Below, we give one of our experimental result set , that covers all of the pattern cases including special cases. Table 2 shows the result set for a given initial constraint, namely *C* and highlighted rows represent the user selected rows among them. Then at the bottom of the table the reformulated constraint *R(C)* is presented for the current user preferences to be used whenever the same user logs on the system with the same input.

Table 2 The experimental result set of a query with initial constraint:

C: Depth Of Coverage = to teach core skills of the domain

Intention	Difficulty	Duration	Cross-Curricular Elements	Required Skills	Covered Skills	Learning Resource Type	Required Degree Of Collaboration
explore	low	about an hour	No	D001-S01	D001-S03	Diagram	with the assistance of teacher
explore	low	about an hour	yes	D001-S02	D001-S11	Figure	alone
explore	low	leisurely	no	D001-S05	D001-S04	Experiment	with a group of students
revise	medium	short	no	D001-S02	D001-S07	Problem Statement	with the assistance of teacher
revise	high	leisurely	no	D001-S01	D001-S11	Diagram	alone
revise	low	short	no	D001-S03	D001-S09	Narrative Text	as a pair of students
explore	medium	about an hour	no	none	D001-S10	Experiment	as a pair of students
explore	high	about an hour	yes	D001-S02	D001-S06	Narrative Text	with the assistance of parents
explore	medium	short	no	D001-S02	D001-S08	Figure	alone
prepare exam	high	leisurely	yes	D001-S04	D001-S11	Narrative Text	with the assistance of teacher
prepare exam	high	leisurely	yes	D001-S04	D001-S10	Narrative Text	with the assistance of teacher

R(C): (Depth of Coverage=to teach core skills of the domain)

AND
(Duration=leisurely)
AND
(Difficulty!=medium)
AND
[[(Intention=prepare exam) AND (Difficulty=high) AND (Cross-Curricular Elements=yes) AND (Required Skills=D001-S04) AND (Learning Resource Type=Narrative Text) AND (Req. Degree Of Collaboration=with the assistance of teacher)] OR [(Covered Skills=D001-S11) AND (Difficulty=high)] OR [(Cross-CurricularElements!=yes) AND (Required Skills!=D001-S04) AND (Learning Resource Type!=Narrative Text) AND (Req. Degree Of Collaboration!=with the assistance of teacher) AND [(Intention=prepare exam)OR(Cross-Curricular elements=no)]]]

4. Conclusion

In this paper, we proposed a personalized heuristic based approach to query formulation. The results are shown on several relational databases covering educational context.

Acknowledgements This work is supported by EC within FP6 under Grant 507922 with the acronym iClass.

References

[1] K.-W. Lai, K.Pratt, A.Grant, State of the Art and Trends in Distance, Flexible, and Open Learning: A Review of the Literature, Submitted to: Distance Learning Reference Group University ofOtago, May 2003
[2] B.G.Wilson, Trends and Futures of Education: Implications for Distance Education, October 2001, received from http://carbon.cudenver.edu/~bwilson/TrendsandFutures.html
[3] e-Learning Interoperability Standards, Sun White paper, January 2002, www.sun.com/products-n-solutions/edu/whitepapers/index.html
[4] IEEE (2002) .Draft Standard For Learning Object Metadata (IEEE P1484.12.1)
[5] ADL. Sharable content object reference model (SCORM) 2004.Technical report, Advanced Distributed Learning, 2004.
[6] P.Ziegler, K.R.Dittrich, Three Decades of Data Integration – All Problems Solved?, First International IFIP Conference on Semantics of a Networked World (ICSNW 2004)
[7] C.Comito, D.Talia, Data integration and query reformulation in service-based Grids: Architecture and Roadmap. CoreGrid Technical Report Number TR-0013, September 6, 2005
[8] C.-C. K. Chang and H. Garcia-Molina, Mind Your Vocabulary: Query Mapping Across Heterogeneous Information Sources, Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data, June 1-3, 1999, Philadelphia, Pennsylvania, USA, pp.335-346
[9] Zhen Zhang, Bin He, and Kevin C.-C. Chang. On-the-fly Constraint Mapping across Web Query Interfaces, In Proceedings of the VLDB-IIWeb'04 Workshop, Toronto, Canada, August 2004
[10] H.-H. Do and E. Rahm, COMA- A system for flexible combination of schema matching approaches, Proceedings of the 28th VLDB Conference, Hong Kong, China, 2002
[11] iClass Inyegrated Project, Annex 1,Description of Work (2004), http://www.iclass.info
[12] laspalmas1.ppt, Information Systems: Modelling Complexity with Categories, Four lectures given by Nick Rossiter at Universidad de Las Palmas de Gran Canaria, 15th-19th May 2000, under the Socrates-Erasmus Programme
[13] Federated Database Systems.ppt, Federated Database Systems Part I, CSCI 8370 Advanced Database, Meena Nagarajan, received from http://lsdis.cs.uga.edu/~meena/Spring06/ADB/1