Chapter 9

Approximate Query Answering with Knowledge Hierarchy

Wookey Lee
Inha University, Korea

Myung-Keun Shin
Telecom Business Division, SK C&C, Korea

Soon Young Huh
Korea Advanced Institute of Science and Technology, South Korea

Donghyun Park
Inha University, South Korea

Jumi Kim
Small Business Institute, Korea

ABSTRACT

Approximate Query Answering is important for incorporating knowledge abstraction and query relaxation in terms of the categorical and the numerical data. By exploiting the knowledge hierarchy, a novel method is addressed to quantify the semantic distances between the categorical information as well as the numerical data. Regarding that, an efficient query relaxation algorithm is devised to modify the approximate queries to ordinary queries based on the knowledge hierarchy. Then the ranking measures work very efficiently to cope with various combinations of complex queries with respect to the number of nodes in the hierarchy as well as the corresponding cost model.

INTRODUCTION

Database query processing has mostly focused on addressing exact answers in terms of Boolean model. There are a number of circumstances in which a user desires an approximate answer rather than the exact answer. At first, when a user does not always understand all about the data schema or the queries contain errors syntactically or semantically, then the query results may be null or be thrown up too much.
Then the user feels to amend or modify the query. Secondly, in data mining environment, when an initial query is answered and that can be considered as an anchor point from which the query can be relaxed to find more detailed information. Manual relaxation, however, for the unsatisfactory queries is usually a drudgery and time-consuming process, which strongly requires a knowledge-based schema for the database or datawarehouse as well as query relaxation mechanism.

The query relaxation process can be explained in more detail by the following example: Consider an illustrative recruiting scenario in which the query:

\[ Q: \text{Skill} = \text{‘C++’} \land \text{Salary} = $40,000 \land \text{Age} = 40 \]

Assume that no result record comes out with the conventional query answering systems. Then in our approach, the first step to relax the query condition is as follows:

\[ Q_r: \text{Skill in (‘Cobol’ ‘C++’ ‘Java’) } \land \text{ $35,000 \leq \text{Salary} \leq 45,000 \land 37 \leq \text{Age} \leq 43$.} \]

And then, we sort the relaxed query results in terms of a ranking measure between the original query and the objects, which will prove very useful for the applicants as they obtain a richer result of information. Finally, we get the results sorted by ranking distance \( D \), such as (1) < Martin, C++, $40000, 40, D: 0.00 >, (2) < Albert, Java, $43000, 40, D: 0.10 >, (3) < Harry, C++, $37000, 38, D: 0.21 >, and (4) < Neal, Cobol, $38000, 41, D: 0.39 >. \]

In order to achieve this, a method of obtaining the approximate value and to measure the distance between the target value and the approximate value needs to be provided. For the numerical domain, such as Salary and Age, the difference between two values can be used as a semantic distance measure. For the categorical domain such as Skill, the approximate values can be calculated by using a predetermined item distance table (Motro, 1990) or by the abstract hierarchy (Chu et al., 1996; Chen, Zhou, & Zhang, 2006).

The approaches based on the semantic distance approach (Motro, 1990; Muslea, 2004; Lee et al., 2007) uses the notion of semantic distance to represent the degree of similarity between data values. Since query answering systems employing the semantic distance approach provide quantitative measures between target values and neighborhood values as a query result, users can retrieve approximate values more effectively using the measures as references to compare with different approximate values. However, for categorical data, the semantic distance approach has two problems because it employs a two dimensional table to store distances among all pairs of data values. First, to find neighbor values of a target value, the system has to scan all the records related to the target value. Second, when a new value is added to a domain, it is required to consider distances between the value and all existing attribute values. This task contains a large amount of overhead to be done by a human operator, and moreover, human operators are liable to lose consistency in assigning distance data to a large number of values. In contrast, the approaches based on the abstraction hierarchy are suitable to dealing with categorical data. However, abstraction approaches could not properly handle other data types, such as the number, money, date and time, etc, and do not provide quantitative similarity measure among data values.

To overcome these problems, we propose a hierarchical quantified knowledge (HQK) that integrates abstraction approach and semantic distance approach. The HQK uses the hierarchy structure of abstraction approach and provides a quantitative measure between data values in the hierarchy. The abstraction hierarchy facilitates finding neighbor values for a target value quite easily. The distance information embedded in the HQK provides a more efficient method than the one based on a table. Maintenance of distance information due to the addition of a new value can be minimized since the change is localized in the hierarchy. This paper will demonstrate how