Cooperative Query Answering Using the Metricized Knowledge Abstraction Hierarchy

Shin, Myung Keun*

Abstract

Most conventional database systems support specific queries that are concerned only with data that match a query qualification precisely. A cooperative query answering supports query analysis, query relaxation and provides approximate answers as well as exact answers. The key problem in the cooperative answering is how to provide an approximate functionality for alphanumeric as well as categorical queries. In this paper, we propose a metricized knowledge abstraction
hierarchy that supports multi-level data abstraction hierarchy and distance metric among data values. In order to facilitate the query relaxation, a knowledge representation framework has been adopted, which accommodates semantic relationships or distance metrics to represent similarities among data values. The numeric domains also compatibly incorporated in the knowledge abstraction hierarchy by calculating the distance between target record and neighbor records.

Keyword: 협력적 질의처리(Cooperative Query Answering), 질의결과 순위결정(Ranking Query Results), 추상화 계층(Abstraction Hierarchy).

1. Introduction

Conventional database systems usually get a query and generate the corresponding result. If the input query, however, has no answer, or the data is not available, or the query is not well-formed with respect to the database schema, a null answer or an error will be given. A cooperative query answering [2, 4, 5, 13, 14] can provide relevant information, permitting conceptual level queries of a wider scope or even approximate answers. To facilitate the query relaxation and the approximate information search, a knowledge representation framework has been one of the most important factors that have influence on performance the cooperative query answering system. A variety of studies on knowledge representation has been performed using logic models, semantic distance, abstraction, and so on.

The logic model approach [3, 8, 11] uses first-order logic predicates to represent the semantic relationship and integrity constraints among data values. Thus, the entire database consists of a set of base predicates. A database query is also written by a predicate rule whereby inquired information is specified with free variables. The query is answered through conflict resolution and inference mechanism. However, this approach has limitations in guiding the query relaxation process and the less intuitive query answering process due to a lack of systematic organization. Therefore logic model approach is not adequate to build up a large scaled system.

The semantic distance approach [7, 12] uses the notion of semantic distance to represent the degree of similarity between data values. Every pair of data values within the data set is supposed to have semantic distances [12], and thus this approach provides straightforward and efficient method for query relaxation and provides ranked results sorted by the semantic distance. The ranked results help the user to find useful information among the results. For categorical data, the distance between two data values is stored in a table. However, since every pair is supposed to have semantic distances, the table size usually gets exploded when realistic application domain is considered, and thus as the table size gets larger, the maintenance cost becomes higher and the consistency of distance measure becomes harder to retain.

The abstraction approach uses the data abstraction method that has been considered as an effective method to accommodate the semantic relationships among data values [1]. In cooperative query answering, such data abstraction is also useful to associate data values for query relaxation. To validate the abstract concepts comprising the related data values, the type abstraction hierarchy [6] introduced the notions of subsumption, composition and abstraction and offered an integrated view of the type hierarchy with multi-level knowledge abstraction. The knowledge abstraction hierarchy (KAH) [10] extended the type abstraction hierarchy, and it focused capturing value abstraction information with additional abstraction information representing domain abstraction knowledge elicited from the underlying databases. However, these methods don’t provide quantitative similarity measure among data values, which is a common limitation of
the methods employing the abstraction approach. Hence, with the abstraction approach, the users could not decide the importance of the results. Also, since the abstraction approach does not match the system that uses semantic distance approach, this approach is not appropriate to build up a scalable and extendable cooperative query answering system.

In this paper, we propose a knowledge abstraction hierarchy that overcomes the problems discussed in the above. The hierarchy facilitates finding neighbor values for a target value in an easy way. We can obtain wider range neighbor values by choosing a higher-level abstract node of the target value. The distance information embedded in the hierarchy provides a more efficient method than the one based on a table. Since assigning distances are limited only to those small number of values directly linked to the new value, consistency in assigning distance data will be improved. It uses the hierarchy structure of abstraction approach and provides a quantitative measure between data values in the hierarchy.

II. Metricized Knowledge Abstraction Hierarchy

In this section we propose a metricized knowledge abstraction hierarchy that combines abstraction hierarchy with a semantic distance notion.

2.1 Abstraction Hierarchy

The MKAH is a knowledge representation framework that facilitates multilevel representation of data and meta-data for an underlying corporate database using data abstraction. Fig. 1 shows an instance of the MKAH that represents the abstraction information on Deformation. Values constituting the hierarchy may be parts of the underlying database or artificial values added to describe the semantic relationship among the existing data values. The MKAH is composed of two types of abstraction hierarchies: value abstraction hierarchy and domain abstraction hierarchy [10]. In the value abstraction hierarchy, there are abstraction relationships of specific node/abstract node. One node in a level can be generalized into an abstract node placed in an upper level. This abstraction relationship can be interpreted as IS-A relationship. For instance, Rolling is a (branch of) Forming while Forming is a (branch of) Deformation. As such, higher levels provide a more generalized data representation than lower ones and the root node can be interpreted as the most abstract, but representative name of the hierarchy. In Fig. 1, the root node is Deformation which can act as a representative of the hierarchy.

The leaf nodes are given with the level value 1, and the level value increases by one at a time when they are generalized with an abstract node. The $n$ level abstract node of a specific node is the abstract node that is located in $n$ level higher than the specific node. The path from a node to the root node is called an abstraction path, and the path from the root node to the node is called a specification path.

![Image of Deformation hierarchy](image)

**Fig. 1.** Metricized knowledge abstraction hierarchy example on Deformation.

**Definition 1 (Level difference)** The level difference value $n$ between two arbitrary nodes is defined as the larger number of abstracted levels from the two nodes to their lowest common abstract node.

**Definition 2 (Neighbor nodes)** The $n$ level neighbor nodes are the nodes that share a common abstract node
in level difference $n$. (See Fig. 2)

Fig. 2. Level difference and neighbor nodes.

2.2 Definition of Distance and Its Calculation in the Hierarchy

In this section we represent the concept of the distance metric and definition of the shortest pass, and define the distance which satisfies the requirement of the distance metric. In adopting the distance metric into the MKAH, we should consider the following properties concerning the distances between nodes in the MKAH:

**Property 1.** Distances between every pair of nodes can be derived.

**Property 2.** The smaller the level difference, the more similar the two nodes are semantically.

Property 1 means that the distance between two arbitrary nodes in the MKAH should be provided. Property 2 means that the smaller the level difference is, the closer the distance should be. The smaller the distance, the more similar the two nodes are semantically. For example, the distance between Forging and Extrusion which have the level difference 1 should be closer than the distance between Forging and Die Casting which have the level difference 2.

Fig. 3. Metricized knowledge abstraction hierarchy on Manufacturing.

To develop the distance metric that satisfies the properties above mentioned, first, we assume basic distance. The basic distance is a real number greater than 0 and not exceeding 1, and the distances are assigned by the domain expert’s own decision. Of course, the basic distances are automatically derived from underlying database using the methods of data mining, neural network, and so on [9]. However, we do not focus on automatic distance evaluation in this paper. The basic distance is specified on the two types of links, vertical link and horizontal link. The vertical link connects a specific node and its 1 level abstract node, and the horizontal link connects two different 1 level neighbor nodes. Fig. 3 shows an example of the MKAH with the basic distance.

**Definition 3 (Shortest path)** The shortest path in the MKAH is defined as the path that has the smallest number of links connecting the nodes.

We can think the shortest path as a concatenation of the following three sub-paths: one abstraction path, one horizontal link and one specification path, where each sub-path can be null. If the horizontal link is not null, this link connects the abstraction
path and the specification path, and the shortest path goes through two 1 level specific nodes of the lowest common abstract node.

Herein, we define the distance between two arbitrary nodes that satisfies the requirement of the distance metric. It is possible to consider only the basic distances on the shortest path for distance calculation however, this approach cannot always guarantee the property 2, implying that sometimes the calculated distance between the two nodes having the level difference 3 might be closer than those having the level difference 2. Therefore, the distances between two arbitrary nodes in the MKAH are formulated using the level difference and the basic distances on the shortest path.

**Definition 4 (Distance)** The distance between two arbitrary nodes $x$ and $y$ in the MKAH, $D(x,y)$, is defined as

$$ D(x,y) = \frac{\sum_{i} \text{bd}(Z_i, Z_{i+1})}{r+1}. $$

where, $\text{bd}(Z_i, Z_{i+1})$ is the basic distance of $Z_i$ and $Z_{i+1}$, and $Z_0, Z_1, \ldots, Z_{r+1}$ is the shortest path of $x$ and $y$.

**Lemma 1.** For two different nodes $x$ and $y$ in the MKAH, the distance of definition 4, $D(x,y)$, is ranged according to the level difference $n$, as

$$ n - 1 < D(x,y) \leq n, $$

where the $n$ is the level difference of $x$ and $y$.

**Proof:** In the equation in Definition 4, since the basic distance is a real number greater than 0 and not exceeding 1, the range of the third term

$$ \frac{\sum_{i} \text{bd}(Z_i, Z_{i+1})}{r+1} $$

is greater than 0 and not exceeding 1. Hence, the distance between two nodes with level difference $n$, $D$, is ranged as

$$ n - 1 < D(x,y) \leq n. $$

**Lemma 2.** For arbitrary nodes $x$, $y$ and $z$ in the MKAH, if the level difference of $x$ and $y$ is smaller than the level difference of $x$ and $z$, then,

$$ D(x,y) < D(x,z). $$

**Proof:** Let $m$ be the level difference of $x$ and $y$, and $n$ be the level difference of $x$ and $z$, then $m - 1 < D(x,y) \leq m$ and $n - 1 < D(x,z) \leq n$. Since $m < n$,

$$ D(x,y) < D(x,z). $$

The distance of Definition 4 satisfies the requirement of the distance metric. The distance between two arbitrary nodes can be determined, and the distances are grouped with respect to the level difference (Lemma 1 and 2).

### 2.3 Data Model and Operations

In this section we present the data model and several operations to manage the MKAH. Fig. 4 shows four

<table>
<thead>
<tr>
<th>domain</th>
<th>abstract_domain</th>
<th>MKAH</th>
<th>level</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAJOR GROUP</td>
<td>NULL</td>
<td>Major &amp; Skill</td>
<td>4</td>
</tr>
<tr>
<td>MAJOR</td>
<td>MAJOR GROUP</td>
<td>Major &amp; Skill</td>
<td>3</td>
</tr>
<tr>
<td>EXPERTISE</td>
<td>MAJOR</td>
<td>Major &amp; Skill</td>
<td>2</td>
</tr>
<tr>
<td>SKILL</td>
<td>EXPERTISE</td>
<td>Major &amp; Skill</td>
<td>1</td>
</tr>
</tbody>
</table>

**ABSTRACTION**

<table>
<thead>
<tr>
<th>specific_node</th>
<th>domain</th>
<th>abstract_node</th>
<th>distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deformation</td>
<td>MAJOR</td>
<td>Manufacturing</td>
<td>0.5</td>
</tr>
<tr>
<td>Forming</td>
<td>EXPERTISE</td>
<td>Deformation</td>
<td>0.3</td>
</tr>
<tr>
<td>Casting</td>
<td>EXPERTISE</td>
<td>Deformation</td>
<td>0.4</td>
</tr>
<tr>
<td>ROLLING</td>
<td>EXPERTISE</td>
<td>Forming</td>
<td>0.5</td>
</tr>
</tbody>
</table>

**HORIZONTAL_DISTANCE**

<table>
<thead>
<tr>
<th>domain</th>
<th>node1</th>
<th>node2</th>
<th>distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAJOR</td>
<td>Deformation</td>
<td>Machining</td>
<td>0.4</td>
</tr>
<tr>
<td>EXPERTISE</td>
<td>Forming</td>
<td>Casting</td>
<td>0.5</td>
</tr>
<tr>
<td>SKILL</td>
<td>Rolling</td>
<td>Forming</td>
<td>0.5</td>
</tr>
</tbody>
</table>

**ATTRIBUTE_DOMAIN_MAPPING**

<table>
<thead>
<tr>
<th>relation</th>
<th>attribute</th>
<th>domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEE SKILL</td>
<td>skill</td>
<td>SKILL</td>
</tr>
<tr>
<td>COLLEGE MAJOR</td>
<td>major</td>
<td>Major</td>
</tr>
<tr>
<td>SKILL FOR TASK</td>
<td>required skill</td>
<td>SKILL</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Fig. 4. Example Knowledge database for managing the Manufacturing MKAH.

그림 4. Manufacturing MKAH를 관리하기 위한 자식 데이터베이스 예제

*Bạn đã đọc nội dung đúng như mong muốn?*
relations (DOMAIN_ABstraction, ABSTRACTION, HORIZONTAL_DISTANCE and ATTRIBUTE_DOMAIN_MAPPING), which are a knowledge database. Using these relations, we consider the following two operations.

1. ABSTRACTNODE(n, D, l) returns the l level abstract node of node n in domain D. If n is the root, which has no parent, null is returned. This function refers the relation ABSTRACTION and DOMAIN_ABSTRACTION to find the abstract node.

2. SPECIFICNODE(n, D, l) returns the l level specific nodes of node n in domain D. If n is the leaf, which has no child, null is returned. This function refers the relation ABSTRACTION and DOMAIN_ABSTRACTION to find the specific nodes.

III. Cooperative Query Answering Using MKAH

The MKAH performs query relaxation that can be invoked in various ways depending on the user’s need and knowledge of the MKAH. In the MKAH, an abstract node and its specific nodes are considered to be conceptually equal since they have an IS-A relationship. On the other hand, neighbor nodes are approximately equal since they have the same abstract node that is conceptually equal to each neighbor nodes (Fig. 5). We present four typical query answering mechanisms: approximate selection, approximate join, conceptual selection, and conceptual join. The queries posed to an underlying database are classified according to these relaxation methods of the MKAH.

Fig. 5. Two kinds of equality for relaxing search condition.

We use similar-to operator, of which symbol is ‘=?’ to express cooperative queries (6, 12). Cooperative queries are specified simply by using ‘=?’ in the where clause of the SQL statement. For the demonstration and explanation of the query answering processes, we use a simplified personnel database that is defined as the following:

EMPLOYEE (id, name, dept, title)
EMPLOYEE_SKILL (id, skill, level)
COLLEGE_MAJOR (id, major, entrance_date, graduate_date)
SKILL_FOR_TASK (task, required_skill)
MAJOR_FOR_TASK (task, required_major)

The EMPLOYEE relation provides the current job position information of an employee, while the EMPLOYEE_SKILL relation provides the skill of employee. SKILL_FOR_TASK relation defines prerequisite skill for tasks, and MAJOR_FOR_TASK relation prescribes the relationships between individual tasks and the college major requirements for the task.

First, an approximate selection provides not only the target value itself but also its approximately equal values. For example, if the right candidate employees who have the skill on Milling and Welding are unavailable or there are not enough, other employees with related skills need to be obtained by enlarging the scope of the search. The approximate selection query for searching for employees with the skill on Milling and Welding is written as follows:
Original Cooperative Query

Select e.name, e.dept
From EMPLOYEE e, EMPLOYEE_SKILL s
Where s.skill = 'Milling' and
s.skill =? 'Welding' and
e.id = s.id

Query Relaxation

Step 1.

where s.skill is-a
ABSTRACTNODE('Milling', 'SKILL', 1)
and s.skill is-a
ABSTRACTNODE('Welding', 'SKILL', 1)
and e.id = s.id
⇒ where s.skill is-a 'Removing' and
s.skill is-a 'Joining' and
e.id = s.id

Step 2.

where s.skill in
SPECIFICNODE('Removing', 'EXPERTISE', 1)
and s.skill in
SPECIFICNODE('Joining', 'EXPERTISE', 1)
and e.id = s.id
⇒ where s.skill in 'Grinding', 'Milling'
and s.skill in ('Welding', 'Bolting')
and e.id = s.id

Note that if the similar-to operator is to be meaningful, both the attribute and specified value should be in the same domain. In this sense, the approximate selection in the above query is valid since both domains are identically SKILL. The original cooperative query can be relaxed by using two operations ABSTRACTNODE() and SPECIFICNODE() defined in section 2.3. In step 1, target value Milling and Welding are generalized to Removing and Joining respectively by using the operation ABSTRACTNODE(). In step 2, by using the operation SPECIFICNODE(), the query condition is relaxed with (Milling, Grinding) and (Welding, Bolting) and the relaxed query can be answered as an ordinary SQL query. As a result of the relaxed query, the system will return the employees who have the requisite skill in Removing and Joining not only the employees who have the skill in Milling and Welding. Also they are ranked according to the semantic distance that is calculated using the equation

$$SD = \sum_{i=1}^{n} w_i \cdot D(v_i, v)$$

where SD is a semantic distance between the target condition and the relaxed condition, and n is the number of the similar-to operator in the query, and w_i is the weight value for each condition, and t_i and r_i are the target value and relaxed value of each condition.

Second, as an extension of the approximate selection, an attribute in the join condition can be abstracted into an approximate range of nearby values. If viewed in terms of the MKAH, the query is equivalent to joining the two attributes on the basis of their abstract values, which would bring in broader join results than the join based on the ordinary specific values. The approximate join query searching for people having experienced a skill that is close to the required skill for Manufacturing Manager task is written as follow:

Original Cooperative Query

Select e.name, e.dept, e.title
From EMPLOYEE e, EMPLOYEE_SKILL s,
SKILL_FOR_TASK t
Where e.id = s.id and
t.task = "Software Design" and
s.skill =? t.required_skill

Query Relaxation

Step 1.

From EMPLOYEE e, EMPLOYEE_SKILL s,
SKILL_FOR_TASK t
Where e.id = s.id
and t.task = "Manufacturing Manager"
and ABSTRACTNODE(s.skill, 'SKILL', 1)
= ABSTRACTNODE(t.required_skill, 'SKILL', 1)

Step 2.

from EMPLOYEE e, EMPLOYEE_SKILL s,
SKILL_FOR_TASK t,
ABSTRACTION a1, ABSTRACTION a2
Where e.id = s.id
and t.task = "Manufacturing Manager"
and s.skill = a1.SPECIFIC_NODE
and t.required_skill= a2.SPECIFIC_NODE
and a1.ABSTRACT_NODE= a2.ABSTRACT_NODE

The use of the similar-to operator is quite analogous to the case in the approximate selection query in the sense that both compared attributes are to be in the same domain. In the above example both domains are SKILL. In step 1, query relaxation is made by abstracting the two attributes using the ABSTRACTNODE() operation. Strictly speaking, the ABSTRACTNODE() operation could not be applied to an attribute, but we just use the operation as an intermediate process to explain the query relaxation. In step 2, Since the ABSTRACTION relation provides pair of specific value and abstract value, joining the two relations on the basis of common abstract nodes can be performed using the ABSTRACTION relation as an intermediary. A relaxed ordinary query can be written as the query in step 2. Also, The distance between s.skill and t.required_skill would be calculated and the results are ranked by the semantic distances.

The conceptual selection and the conceptual join can be handled similar way to handle the approximate selection and the approximate join.

IV. Implementation

We implemented the MKAH to construct a career job search system that considers skill, salary and age of applicants. This system employed the Microsoft SQL Server 2000 as the database server, and the user interfaces were programmed with Microsoft Visual C++ 6.0. Fig. 6 shows the example of the cooperative query answering system. 'rolling' is selected as the target value of skill domain, and '70' is selected as the target value of the salary domain, and '30' is selected as the target value of the age domain. This will perform an approximately equal 1-level search on the skill domain and range queries for salary and age domain. The query returns the cooperative answers and the distance from target values to each answer is calculated using the equation (1). We use values 1, 10 and 3 for each skill, salary and age domain as radius value.

Fig. 6. Example of metricized cooperative query answering.

그림 6. 개념화된 협력적 질의 처리 예

V. Conclusions

In this paper, we have proposed the MKAH as a knowledge representation framework to support cooperative query answering which relaxes the search condition and provides approximate answers as well as exact answers. The MKAH has an abstraction-based hierarchy that supports the multi-level knowledge representations, and manages the basic distances among neighbor nodes. We have defined the distance metric that calculates distances between two arbitrary nodes by incorporating the basic distances. The calculated distances are grouped by the level differences, and they are precisely discriminated according to the similarity, so that we can get consistency in the distance metric among all nodes in the MKAH. Since the MKAH considers only neighbor values when managing the distances, while the tabular methods consider all the other values, maintenance cost is considerably reduced with the MKAH.
In terms of cooperative query answering, the MKAH can guide more interactive and flexible query relaxation with the abstraction level information. Since the MKAH supports the distance metric, it is appropriate to handle the quantitative similarity of categorical data, and corresponding query results are ranked according to the distances from query conditions. Also, the domain adopting the MKAH could be compatible with other numeric domains when calculating the distance between target record and neighbor records.

Finally, we have presented the data model and operations to implement the MKAH, and have implemented a prototype cooperative query answering system to evaluate the effectiveness of the proposed framework. Our preliminary experimental results reveal that the MKAH provides rich semantic representation and a high quality distance metric and is appropriate for building a scalable and extendable system by integrating the semantic distance approach.

The MKAH can be appropriate for a knowledge representation framework of the systems manipulating nominal data as well as numeric data, such as, knowledge management systems, search engines, decision support systems and so on.

For future research, systematic procedures to construct the MKAH should be conceived for practical application. In addition, we will study the context sensitive MKAH which modulate the basic distance values as context changes.

Reference


