Discovering Database Summaries through Refinements of Fuzzy Hypotheses

Do Heon LEE and Myoung Ho KIM
Department of Computer Science
Korea Advanced Institute of Science and Technology
Taejon, 305-701, South Korea
{dhlee, mhkim}@adam.kaist.ac.kr

Abstract

Recently, many applications such as scientific databases and decision supporting systems that require comprehensive analysis of a very large amount of data, have been evolved. Summary discovery techniques, which extract compact representations grasping the meanings of large databases, can play a major role in those applications. In this paper, we present an effective and robust method to discover simple linguistic summaries. We first propose a hypothesis refinement algorithm that is a key technique for our summary discovery method. Using the algorithm, a formal procedure for summary discovery is presented together with an illustrative example. Our discovery method can handle both rigid concepts and fuzzy concepts that occur frequently in practice. Discovered summaries can also be regarded as high-level interattribute dependencies.

1 Introduction

Summary discovery is one of the major parts of data mining techniques, which provide the user with comprehensible representations for a large amount of data. Data summaries, i.e., general data characteristics, provide means of answering particular questions about the data and ways of formatting the information to enable an analyst to comprehend the contents of the data easily[1]. They also provide a starting point for the ability to make useful inferences from large collections of data. The summary that “Most students in biology department prefer natural science subjects” allow a curriculum planner to make inferences about the viability of opening either “chemistry” or “physics” course in the next term.

Several requirements for effective summary discovery techniques from actual databases must be considered. First, since most databases contain nominal data as well as numerical data, discovery techniques must be able to deal with both types of data. Second, those techniques must be robust in a sense that some noisy data can not affect the major results. It is because actual databases usually contain noisy data due to either wrong inputs or genuine exceptions[2]. Third, the discovered summaries must have simple forms, because summary discovery is done to comprehend a large amount of data more easily. Four, an user can specify focused data for summarization using vague terms. Since summary discovery facilities are expected to be mainly used in explorative environments, an user is apt to have difficulties in specifying his focused data precisely before knowing global data configurations.

Recently, there have been several database-oriented researches on summary discovery techniques[1, 3, 4, 5]. They commonly use various forms of concept trees/hierarchies that represent relationships among various domain concepts. In fact, exploitation of such domain knowledge is a crucial component of knowledge discovery in databases[2]. Each node in a concept tree represents a concept in the problem domain. The ancestor-descendant relationships among nodes in the tree correspond to general-specific relationships among concepts. All the techniques in [3, 4, 5] assume that each concept has a rigid boundary. In other words, when several elements constitute a concept, each element either belongs to the concept completely, or does not belong to it at all. However, there are many exceptions in real world. For instance, since the applied statistics may have some aspects of engineering, it is hard to say that the applied statistics is a pure natural science subject. The argument that the applied statistics is not a natural science subject is not acceptable either. The following example illustrates how the rigid boundary problem makes difficulties in summary discovery.

Suppose that Figure 1 is an enrollment relation in a biology department. What is a correct answer to the question, “Which subject between the natural science and the engineering is preferred by students in the biology department?”. The answer may not be intuitively clear. This is because the applied statistics does not completely belong...
to either of the natural science and engineering.

In this work, we propose a hypothesis refinement algorithm for constructing and evaluating fuzzy hypotheses, i.e., actually hypothetical summaries. The algorithm utilizes hierarchies of domain concepts with fuzzy boundaries rather than rigid ones. Using the algorithm, we present an effective and robust procedure to discover simple linguistic summaries, which accommodate nominal data as well as numerical data. Our method employs the flexibility of a strictly extended relational data model, so called Level-1 Fuzzy Relational Data Model (FRDM-1) [6], where user's vague requests for focused data can be easily expressed.

The rest of the paper is organized as follows. In Section 2, FRDM-1 is briefly described for self-explanatory purpose. Section 3 proposes the hypothesis refinement algorithm that is a key technique of our summary discovery method. Section 4 presents the overall procedure of the proposed method and provides an example of discovering summaries. Section 5 concludes and mentions the future research problems.

2 The Level-1 Fuzzy Relational Data Model (FRDM-1)

Note that our discovery procedure which permits vague specification of data portions to be summarized, is conceived for applying to ordinary relational databases. Thus, it is needed to choose an extended data model to support vague query specification, while having sufficient compatibilities with relational databases. Among many fuzzy database proposals for vague queries including [8, 9], FRDM-1 is directly applicable to ordinary relational databases through its Crisp-Database-Fuzzy-Query approach[6]. In addition, it provides a theoretical background for various fuzzy query languages.

Basic objects in FRDM-1 are fuzzy relations. They have the same forms as ordinary data relations except the augmented \( \mu \) attributes, which represent membership degrees of individual tuples. The schema of fuzzy relations is as follows.

\[
\hat{R} = \{(t, \mu_R(t)) \mid t = (x_1, \ldots, x_n), 0 < \mu_R(t) \leq 1, x_1 \in D_1, \ldots, x_n \in D_n \},
\]

where \( \mu_R(t) \) denotes the degree to which tuple \( t \) belongs to the fuzzy relation \( \hat{R} \).

If we restrict the value of \( \mu_R(t) \) to be either 0 or 1, a fuzzy relation is reduced to an ordinary relation. The relational algebra and the relational calculus are extended to the Level-1 Fuzzy Relational Algebra (FRA-1) and the Level-1 Fuzzy Relational Calculus (FRC-1) by utilizing the extension principle of the fuzzy set theory[10]. Likely to the conventional model, they have equivalent expressiveness to each other.

The Fuzzy Selective Relational Algebra (FSRA), which is a further extension of FRA-1 is defined to facilitate expressing vagueness in data requests. While other operators of FSRA are the same as those of FRA-1, the selection operator is further extended to accommodate fuzzy constants and fuzzy comparators. The precise definition of selection operator of FSRA is as follows.

\[
\sigma_{X \Theta Y}(\hat{R}) = \{(t, \mu_{X \Theta Y}(\hat{R})(t)) \mid \mu_{X \Theta Y}(\hat{R})(t) = t - \text{norm}((\mu_R(t), v(X \Theta Y)[t])),
\]

where \( X \) is an attribute name and \( \Theta \) is a comparator among \( =, \neq, \leq, >, <, \approx \) and \( ! \approx \). The fuzzy comparators \( \approx \) and \( ! \approx \) represent 'similar-to' and 'dissimilar-to' comparisons, respectively. \( Y \) is either an attribute name or a constant. Here the constant is either a crisp constant or a fuzzy constant. \( v(X \Theta Y)[t] \) denotes conformance degree of the tuple \( t \) with respect to the fuzzy predicate \( X \Theta Y \)[10].

Every FSRA query can be transformed to combinations of FRA-1 queries. An example of FSRA query is such that \( \sigma_{\text{AGE equal young}}(EMP) \). The term 'young' is an example of fuzzy constant. Such fuzzy constants and fuzzy comparators are useful in expressing vague data requests. Actual meanings of them are maintained in semantic relations, which are in the same forms as those of ordinary data relations.

3 A Hypothesis Refinement Algorithm

A hypothesis is an assertion supposed to be true. In this section, a fuzzy hypothesis is defined and fuzzy hypothesis evaluation method is presented. After providing how such fuzzy hypotheses are constructed through fuzzy hypothesis hierarchies, a precise algorithm for constructing and evaluating fuzzy hypotheses is proposed.

3.1 Fuzzy Hypotheses

Since fuzzy concepts are effective to express complex phenomena in simplified forms [10], we use hypotheses hav-
ing fuzzy concepts, i.e., fuzzy hypotheses, as vehicles to contain hypothetical summaries that are easily comprehensible. A fuzzy concept is represented as a fuzzy restriction, i.e., a fuzzy set[10]. We use the term ‘evaluation’ instead of ‘proving’ to mean the activity of testing the truth of a fuzzy hypothesis, because the truth becomes a fuzzy value rather than either TRUE or FALSE. We use the following notational conventions herein. $A_j$ denotes an attribute name of a fuzzy relation $R$. A fuzzy set $\tilde{F}_j$ is represented by its membership function $\mu_{\tilde{F}_j}(x)$, where $x$ is an element on the domain, i.e., the universe of discourse. Since all relations and hypotheses in this paper are fuzzy relations and fuzzy hypotheses, we will drop the term ‘fuzzy’ for simplicity throughout the paper except the cases where emphases are needed.

Definition 1

A fuzzy hypothesis is defined recursively as follows.

* “$A_j$ is $\tilde{F}_j$” is an atomic fuzzy hypothesis, where $\tilde{F}_j$ is a fuzzy restriction defined on the domain of $A_j$. An atomic fuzzy hypothesis is a fuzzy hypothesis.

* If $H$ and $I$ are fuzzy hypotheses, then the conjunction of $H$ and $I$, i.e., $H \land I$, is also a fuzzy hypothesis.

A fuzzy hypothesis template is a parameterized fuzzy hypothesis where all fuzzy restrictions are variables. A fuzzy hypothesis template is said to be instantiated when its variables are all substituted by specific fuzzy restrictions.

(End of Definition 1)

Examples of fuzzy hypotheses are “$AGE$ is young”, and “($SALARY$ is high) $\land$ ($AGE$ is young)”. An example of fuzzy hypothesis templates is “($SALARY$ is $X$) $\land$ ($AGE$ is $Y$)”. Now, let us see how to evaluate truth values of such fuzzy hypotheses on actual databases. When a data relation $\tilde{R}$ and fuzzy restrictions $\tilde{F}_j$’s are given, we can evaluate the truth of a hypothesis $H$ with respect to the relation $\tilde{R}$. The truth of a hypothesis depends on the degree to which the tuples in the relation $\tilde{R}$ support the hypothesis $H$. We use $SD_{\tilde{R}}(H)$ to denote the support degree of the relation $\tilde{R}$ for the hypothesis $H$. We will omit a relation $\tilde{R}$ in $SD_{\tilde{R}}(H)$, i.e., $SD(H)$, if there is no ambiguity. In what follows, we describe how fuzzy hypotheses are evaluated.

* Atomic fuzzy hypotheses

Since $\mu_{\tilde{R}}(t_i)$ denotes the degree to which the tuple $t_i$ exists in the fuzzy relation $\tilde{R}$ and $\mu_{\tilde{F}_j}(t_i.A_j)$ represents the degree to which $t_i$ supports a fuzzy hypothesis expressed with a fuzzy restriction $\tilde{F}_j$, their t-norm value becomes to represent the degree to which the fuzzy hypothesis is supported by the existence of tuple $t_i$. For example, suppose a fuzzy restriction $\tilde{F}_j$ is given as young = { (10,1.0), (20,0.9), (30,0.8), (40,0.4), ..., (60,0.0) }. When a fuzzy hypothesis “Employees are young” is evaluated on a employee database, an definite existence of a 20 year-old employee supports the hypothesis to degree as much as 0.9, that is the membership degree of ‘20’ in the fuzzy restriction young. Similarly, a fact that a 30 year-old employee exists at possibility 0.7 supports the hypothesis to degree as much as $\min(0.7, 0.8) = 0.7$, when MIN operation is supposed to reflect conjunctive semantics. Then, it is easy to see that the support degree of an atomic hypothesis is the normalized sum of support degrees of all individual tuples.

The support degree becomes to represent the ratio of the number of supporting tuples to the total number of tuples in $\tilde{R}$ semantically. Similar notions have been commonly used in evaluating fuzzy quantifiers[1, 12, 13]. In evaluating conjunctive hypotheses, we use a t-norm operator again to reflect the conjunctive semantics of two fuzzy hypotheses. Note that even if the data relation $\tilde{R}$ is an ordinary relation, i.e., $\mu_{\tilde{R}}(t_i)$’s are all 1’s, the evaluation method can also be used consistently.

A binary operator $\otimes$ belongs to the class of t-norm operators, which are generally used to represent conjunctions in fuzzy logical contexts, if it satisfies the following axiomatic properties[10].

When $a, a_1, b, c \in [0,1],$

1. $\otimes(0,0) = 0; \otimes(a,1) = \otimes(1,a) = a,$
2. $\otimes(a_1, b_1) \leq \otimes(a_2, b_2)$ if $a_1 \leq a_2$ and $b_1 \leq b_2$ : monotonicity,
3. $\otimes(a, b) = \otimes(b,a)$ : commutativity,
4. $\otimes(\otimes(a,b), c) = \otimes(a, \otimes(b,c))$ : associativity.

In addition to those common properties, each of t-norm operator has its own specific properties, which determine the suitability of the operator for a specific application.
Examples of the $\otimes$ operator include MIN and probabilistic product operators [10].

### 3.2 Fuzzy Hypothesis Hierarchies

Now that the evaluation method for fuzzy hypotheses has been presented, we provide how such fuzzy hypotheses are constructed. Fuzzy hypotheses are constructed in a stepwise refinement fashion through fuzzy hypothesis hierarchies. Once we generate a collection of fuzzy hypotheses with the most general terms, we select hypotheses supported strongly by actual database. The selected hypotheses are refined to include more specific terms. To facilitate the refinement procedure, we first define a fuzzy restriction hierarchy composed of concept nodes, each of which represents a concept with fuzzy boundaries.

**Definition 2**

A fuzzy restriction hierarchy is a partially ordered set, $(\Gamma, \subseteq)$ where $\Gamma$ is a set of fuzzy restrictions defined on the domain $D$. For $F_i$ and $F_j$ in $\Gamma$, $F_i \subseteq F_j$ if $\forall x \in D, \mu_{F_i}(x) \leq \mu_{F_j}(x)$. $F_i$ is called a specialization of $F_j$ if $F_i \subseteq F_j$ and $F_i \neq F_j$. $F_i$ is called a maximal fuzzy restriction if there is no other $F_j$ such that $F_i \subseteq F_j$. A rooted fuzzy restriction hierarchy is a fuzzy restriction hierarchy with unique maximal fuzzy restriction.

(End of Definition 2)

For example, if we define membership functions of 'about 50' and 'almost 50' as in Figure 2, 'almost 50' is a specialization of 'about 50'. A fuzzy restriction whose membership degrees are all 1's, so called 'ALL', is uniquely maximal in all possible fuzzy restriction hierarchies defined on the domain $D$. Thus, every non-rooted fuzzy restriction hierarchy can become rooted by augmenting the fuzzy restriction 'ALL' to it. From now on, we consider only rooted fuzzy restriction hierarchies. Based on a fuzzy restriction hierarchy, we define a fuzzy hypothesis hierarchy.

**Definition 3**

A fuzzy hypothesis hierarchy is a partially ordered set $(\mathcal{H}, \preceq)$ where $\mathcal{H}$ is a set of fuzzy hypotheses. For $H_i$ and $H_j$ in $\mathcal{H}$, $H_i \preceq H_j$ iff fuzzy hypothesis templates of $H_i$ and $H_j$ are the same as each other and each fuzzy restriction in $H_i$ is either the same as or a specialization of its corresponding fuzzy restriction in $H_j$. $H_i$ is called a specialization of $H_j$ if $H_i \preceq H_j$ and $H_i \neq H_j$. And, $H_i$ is called a direct specialization of $H_j$ if $H_i \preceq H_j$, $H_i \neq H_j$, and there is no other $H_k$ such that $H_i \preceq H_k \preceq H_j$.

(End of Definition 3)

A fuzzy hypothesis hierarchy is constructed from the given fuzzy restriction hierarchies and a fuzzy hypothesis template. For example, suppose that we are given fuzzy restriction hierarchies in Figures 3, 4 and a fuzzy hypothesis template "(MAJOR is $X$) \land (HOBBY is $Y$)". Figure 5 shows a part of the fuzzy hypothesis hierarchy, where $(\alpha, \beta)$ denotes the hypothesis "(MAJOR is $\alpha$) \land (HOBBY is $\beta$)".

![Figure 3: Fuzzy restriction hierarchies H.MAJOR](image)

![Figure 4: Fuzzy restriction hierarchies H.HOBBY](image)

As shown in Figure 5, even if we have small sets of fuzzy restrictions at first, a large number of hypotheses can be derived due to all possible combinations. However, the following theorem gives an insight by which unnecessary derivations can be avoided.

**Theorem 1** If $H$ is a specialization of $I$, the support degree of $H$ can not be greater than that of $I$. 

226
Figure 5: A part of a fuzzy hypothesis hierarchy

3.3 An Algorithm for Constructing and Supporting Fuzzy Hypotheses

We propose an algorithm for constructing and evaluating fuzzy hypotheses herein. The following algorithm adopts the breadth first search technique with prunings.

A Hypothesis Refinement Algorithm

Given:
- a data relation \( R \)
- fuzzy restriction hierarchies \((\alpha_k, \subseteq_k)\)'s
- fuzzy hypothesis template \( T \)
- a support degree threshold value \( r \)

1. Initially, assign hypotheses made by instantiating the fuzzy hypothesis template \( T \) with \( ALL'S \) in the corresponding fuzzy restriction hierarchies \((\alpha_k, \subseteq_k)\)'s, into \( CS \). And make \( SS \) an empty set.
2. If \( CS \) is an empty set, stop.
3. Make \( NextCS \) an empty set.
4. For each hypothesis \( H \) in \( CS \), do 5-6.
5. Compute support degree \( SD(H) \).
6. If \( SD(H) \) is not less than given threshold value \( r \), append \( H \) into \( SS \) and assign direct specializations of \( H \) in the fuzzy hypothesis hierarchy produced by using \((\alpha_k, \subseteq_k)\)'s and \( T \), into \( NextCS \).
7. Replace \( CS \) with \( NextCS \) and go to 2.

Notations:
- \( CS \) : a set of candidate hypotheses
- \( SS \) : a set of selected hypotheses

The algorithm comes up with a set of fuzzy hypotheses supported more strongly than the support degree threshold value \( r \). In order to utilize this value to identify an ignorable ratio, an hypothesis with a support degree lower than the threshold is of no interest. Thus, if the support degree of a hypothesis is lower than the given threshold value, the further specialization of it is not necessary by Theorem 1.

4 A Summary Discovery Procedure: FuzzySum

The hypothesis refinement algorithm is a powerful mechanism for summary discovery. In order to utilize this
hypothesis refinement algorithm for summary discovery, we need a relevant (focused) data specification, a hypothesis template description and names of preferred fuzzy restriction hierarchies.

4.1 Overview of the Procedure FuzzySum

Though a database has a large amount of data, only some portions of it are relevant to discover summaries. A FSRA query explained in Section 2 can make the discovery procedure focus on only relevant data. Remind that they are similar to normal data retrieval queries except permitting fuzzy constants and fuzzy comparators. By using such fuzzy constants and fuzzy comparators, an user can effectively express his vague focus request for summarization, that are easily found in various explorative applications. A hypothesis template specifies which attributes of the data are to be summarized. Fuzzy restriction hierarchies reflect different subjective cognitions of users. We designed a query language accommodating such specifications [7]. An example of summary query is given in Section 4.2. A summary discovery procedure named FuzzySum consists of three major steps.

Procedure for summary discovery: FuzzySum

Step 1 Identify a set of relevant data.
Step 2 Construct a hypothesis template.
Step 3 Discover summaries by executing SelectHypo().

In Step 1, a FSRA query embedded in the given summary query is executed to obtain relevant data. Since the FSRA query imposes fuzzy qualifications, some tuples are strongly relevant and others are weakly relevant. The degrees of such strongness are represented by membership degrees in the resulting fuzzy relations. In Step 2, when the concerning attributes are \( A_1, \ldots, A_n \), its corresponding hypothesis template becomes \( (A_1 \in X_1) \land \ldots \land (A_n \in X_n) \). In Step 3, those summaries are in fact hypotheses that are constructed and evaluated by the hypothesis refinement algorithm SelectHypo(), where support degrees of those hypotheses are no less than the given threshold value.

4.2 An Example of Summary Discovery

Now, we provide an illustrative example of summary discovery. Suppose that a company database has a relation about employees as in Figure 6. Suppose also that we have a series of fuzzy restrictions on each attribute domain, whose semantics are represented in the form of semantic relations as in Figure 8. While the semantic relation for \( \text{AGE} \) is used for execution of a FSRA query, semantic relations for \( \text{MAJOR} \) and \( \text{HOBBY} \) are used in evaluating fuzzy hypotheses. There can be several ways of maintaining semantics of fuzzy restrictions including algebraic functions, lists and semantic relations [6, 10, 11, 12]. Among them, we use the semantic relation approach proposed in [6], because it maintains the semantics in the same forms as those of ordinary data relations. It implies a benefit that they can be directly accessed with normal data operations. Note that however, this does not imply the semantic relation approach is mandatory for our method. Fuzzy restriction hierarchies were already depicted in Figure 3 and Figure 4 (in Section 3.3). A query for discovering summaries is given in Figure 7. It asks for summaries on major studies and hobbies for young employees.

Figure 6: An example data relation \( \text{EMP} \)

<table>
<thead>
<tr>
<th>NAME</th>
<th>DEPT</th>
<th>AGE</th>
<th>MAJOR</th>
<th>HOBBY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lee</td>
<td>planning</td>
<td>24</td>
<td>computer sci.</td>
<td>walking</td>
</tr>
<tr>
<td>Num</td>
<td>planning</td>
<td>30</td>
<td>accounting</td>
<td>walking</td>
</tr>
<tr>
<td>Park</td>
<td>planning</td>
<td>35</td>
<td>accountancy</td>
<td>accounting</td>
</tr>
<tr>
<td>Jung</td>
<td>planning</td>
<td>40</td>
<td>physics</td>
<td>astronomy</td>
</tr>
<tr>
<td>Noh</td>
<td>planning</td>
<td>45</td>
<td>accountancy</td>
<td>art</td>
</tr>
<tr>
<td>Wang</td>
<td>planning</td>
<td>50</td>
<td>physics</td>
<td>calligraphy</td>
</tr>
<tr>
<td>Hong</td>
<td>sales</td>
<td>55</td>
<td>accountancy</td>
<td>calligraphy</td>
</tr>
<tr>
<td>Yoon</td>
<td>planning</td>
<td>60</td>
<td>computer sci.</td>
<td>tennis</td>
</tr>
<tr>
<td>Choi</td>
<td>planning</td>
<td>65</td>
<td>music</td>
<td>swimming</td>
</tr>
<tr>
<td>Sub</td>
<td>develop</td>
<td>70</td>
<td>computer sci.</td>
<td>music</td>
</tr>
<tr>
<td>Moon</td>
<td>planning</td>
<td>75</td>
<td>history</td>
<td>music</td>
</tr>
<tr>
<td>Kwang</td>
<td>planning</td>
<td>80</td>
<td>accountancy</td>
<td>dance</td>
</tr>
</tbody>
</table>

Figure 7: An example summary query

Step 1. Identify a set of relevant data
A FSRA query corresponding to “FROM EMP WHERE \( \text{AGE} = \) young” in the summary query is \( \sigma_{\text{AGE}=\text{young}}(\text{EMP}) \). The execution of it produces a fuzzy relation in Figure 9. Remind that the \( \mu \) attribute values denotes query conformance of result tuples. It becomes the input data relation \( R \) for the algorithm SelectHypo().

Step 2. Construct a hypothesis template
Since the concerning attributes are \( \text{MAJOR} \) and \( \text{HOBBY} \), the corresponding hypothesis template for the query in Figure 7 is \( (\text{MAJOR} \in X) \land (\text{HOBBY} \in Y) \).

Step 3. Execute SelectHypo().
Let us consider the execution of SelectHypo(). The first refinement step is the first level of the hierarchy in
For **AGE**

<table>
<thead>
<tr>
<th>LOWER</th>
<th>UPPER</th>
<th>P&lt;sub&gt;Y&lt;/sub&gt;</th>
<th>P&lt;sub&gt;V&lt;/sub&gt;</th>
<th>P&lt;sub&gt;M&lt;/sub&gt;</th>
<th>P&lt;sub&gt;W&lt;/sub&gt;</th>
<th>P&lt;sub&gt;YV&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>10</td>
<td>0.8</td>
<td>1.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>20</td>
<td>0.4</td>
<td>1.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>30</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>30</td>
<td>40</td>
<td>0.2</td>
<td>0.9</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>40</td>
<td>50</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>50</td>
<td>60</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>60</td>
<td>70</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>70</td>
<td>80</td>
<td>0.4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>80</td>
<td>90</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>90</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**VY:** very young  
**Y:** young  
**Mi:** medium  
**O:** old  
**VO:** very old

For **MAJOR**

<table>
<thead>
<tr>
<th>NAME</th>
<th>P&lt;sub&gt;SCI&lt;/sub&gt;</th>
<th>P&lt;sub&gt;EC&lt;/sub&gt;</th>
<th>P&lt;sub&gt;MA&lt;/sub&gt;</th>
<th>P&lt;sub&gt;EN&lt;/sub&gt;</th>
<th>P&lt;sub&gt;PH&lt;/sub&gt;</th>
<th>P&lt;sub&gt;EOT&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Korean lit.</td>
<td>1.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
<td>0</td>
</tr>
<tr>
<td>English lit.</td>
<td>1.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
<td>0</td>
</tr>
<tr>
<td>history</td>
<td>0.7</td>
<td>0.4</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
<td>0</td>
</tr>
<tr>
<td>account</td>
<td>0</td>
<td>1.0</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
<td>0</td>
</tr>
<tr>
<td>commerce</td>
<td>0</td>
<td>0.3</td>
<td>1.0</td>
<td>0.3</td>
<td>0</td>
<td>1.0</td>
</tr>
<tr>
<td>mgmt. sci.</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
<td>0.3</td>
<td>0</td>
<td>1.0</td>
</tr>
<tr>
<td>comp. sci.</td>
<td>0</td>
<td>0</td>
<td>0.4</td>
<td>1.0</td>
<td>0.4</td>
<td>0</td>
</tr>
<tr>
<td>mechanics</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
<td>0</td>
<td>1.0</td>
</tr>
<tr>
<td>phys. eng.</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
<td>0</td>
<td>1.0</td>
</tr>
<tr>
<td>physics</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
<td>0</td>
<td>1.0</td>
</tr>
<tr>
<td>chemistry</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
<td>0</td>
<td>1.0</td>
</tr>
</tbody>
</table>

**LJ:** literature  
**EC:** economics  
**MA:** management  
**EN:** engineering  
**PH:** natural sciences  
**EOT:** science & technology

For **HOBBY**

<table>
<thead>
<tr>
<th>NAME</th>
<th>P&lt;sub&gt;anges&lt;/sub&gt;</th>
<th>P&lt;sub&gt;sports&lt;/sub&gt;</th>
<th>P&lt;sub&gt;art&lt;/sub&gt;</th>
<th>P&lt;sub&gt;heavy&lt;/sub&gt;</th>
<th>P&lt;sub&gt;light&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>tennis</td>
<td>1.0</td>
<td>0</td>
<td>1.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>swimming</td>
<td>1.0</td>
<td>0</td>
<td>1.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>gymnastics</td>
<td>1.0</td>
<td>0</td>
<td>0.7</td>
<td>1.0</td>
<td>0.7</td>
</tr>
<tr>
<td>dancing</td>
<td>0</td>
<td>0.9</td>
<td>1.0</td>
<td>0</td>
<td>0.9</td>
</tr>
<tr>
<td>music</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.1</td>
<td>1.0</td>
</tr>
<tr>
<td>calligraphy</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
</tr>
</tbody>
</table>

**Panges:** playing  
**Sports:** heavy sports  
**Part:** art  
**Heavy:** heavy sports  
**Light:** light sports

Figure 8: An example semantic relation for **AGE**, **MAJOR** and **HOBBY**

Figure 10. Support degree for each hypothesis is attached below the corresponding hypothesis. In that step, a hypothesis, "**MAJOR** is sci.&tech. and **HOBBY** is sports" is selected for further refinements. The second and third steps are the second level and the third level, respectively, in Figure 10. The underlined hypotheses are selected for next refinements. As a result, four fuzzy hypotheses are selected by the algorithm as strongly supported ones.

In other viewpoint, each refinement step can be regarded as a hypothesis space partitioning. The diagram in Figure 11 depicts the hypothesis space where horizontal and vertical dimensions represent the domains of **MAJOR** and **HOBBY**, respectively. A fuzzy partition(rectangle) in the space can be mapped into a fuzzy hypothesis. The boundary of each partition is not rigid one, but fuzzy. It represents the fact that each hypothesis has fuzzy concept boundary. Note that the smaller the size of partition is, the more specific the corresponding hypothesis is.

Figure 9: A fuzzy relation after a FSRA selection

Figure 11: An example of the space partitioning by SelectHypo()

The resulting summaries are arranged as follows.
- **MAJOR** is sci. & tech. and **HOBBY** is sports. (0.74)
- **MAJOR** is engineering and **HOBBY** is sports. (0.63)
- **MAJOR** is sci. & tech. and **HOBBY** is heavy sports. (0.62)
- **MAJOR** is engineering and **HOBBY** is heavy sports. (0.62)
Thus, we can conclude that 'young' employees tend to have majored in 'science & technology', especially 'engineering' and like to play (heavy) sports.

5 Concluding Remarks

In this paper, we have proposed a hypothesis refinement algorithm for constructing and evaluating fuzzy hypotheses. The hypothesis refinement algorithm first constructs general hypotheses in terms of most general concepts. It evaluates the hypotheses and then refines strongly supported hypotheses into more specific ones. The refined hypotheses are evaluated again for further refinements. After some stepwise refinements, it comes up with useful specific hypotheses as the results. Based on the hypothesis refinement algorithm, we present an effective and robust procedure to discover linguistic summaries from databases. Our summary discovery method is established in the framework of Level-1 Fuzzy Relational Data Model (FRDM-1) in order to achieve capabilities for supporting vagueness in user's summary requests.

The proposed discovery method accommodates both nominal and numerical data through an unified way. It is robust in a sense that some noisy data due to either wrong inputs or genuine exceptions can not affect the major results. As it utilize concept hierarchies with fuzzy boundaries, more closer approximations to real domain knowledge can be exploited. Furthermore, because it results in simple linguistic descriptions of summaries, it is more effective to comprehend a large amount of data. Fuzzy constants and fuzzy comparators in <relevant data specification> can help users to express vague specifications of focused data easily in data exploration phases.

Discovered summaries can also regarded as high-level interattribute dependencies, i.e., IF relevant data specification THEN a discovered summary at degree SD( ). In the summary result in Section 4.2, the summary "MAJOR is sci.&tech. and HOBBY is sports" with respect to the relevant data specification, "AGE is young", can be interpreted as "IF AGE is young then MAJOR is sci.&tech. and HOBBY is sports at degree 0.74". The degree 0.74 represents how much ratio of tuples supports the interattribute dependency.

There remain some open issues to be addressed. Discovered summaries themselves can be considered as useful information to be stored and managed. However, since they are dependent on original raw databases, some summaries may be obsolete after considerable changes in the raw databases. Thus, methods to maintain consistency between the raw database and the derived summary data set should be addressed. In addition, systematic procedures to elicit domain knowledge such as fuzzy restrictions and fuzzy restriction hierarchies should be conceived for practical applications.

References