WP1 - 4:00

INCOMPLETE INFORMATION IN DATABASE SYSTEMS

Eugene Wong
Department of Electrical Engineering and Computer Sciences
and the Electronics Research Laboratory
University of California, Berkeley, California 94720

1. Introduction

There are numerous situations in which a database cannot provide a precise and unambiguous answer to some of the queries that we wish to pose. The potential sources for the difficulty vary. These include examples such as measurement and recording errors, missing data, incompatible scaling, obsolescence, and data aggretion of one kind or another. Different approaches to this problem have been tried. These range from a consistent way of handling a place-holder "value not known" to Lipski's recent work [1] on dealing with the truth value "possible" in an extended propositional calculus. Although the focus is different, the problem also arises in the artificial intelligence literature (see [2]).

In many of the situations where a precise answer cannot be obtained from the database, much more prior information than "value unknown" or "predicate is possibly true" is available. The goal of this paper is to propose a framework wherein such prior information can be effectively exploited.

The organization of this paper is as follows: First, we shall enumerate a number of commonly occurring sources of imprecision, and propose a general model that encompasses all of these. Using this model, we shall restate queries on an imprecise database as problems of statistical inference. We then propose a definition for "answers" to a query, and consider the merits of these definitions relative to processing ease and consistency under query transformations. Problems of acquisition and storage of a priori statistical information are of great practical importance, but their consideration will be deferred until a follow-up study.

2. Sources of Imprecision

We begin with an enumeration of some common sources of imprecision.

a. Scale Differences. Here, we are referring to scale differences that cause an ambiguity and not merely a change in units. For example, changing a temperature from °F to ° C is a change in units, but changing temperature in °F to one of four values (cold, cool, warm, hot) is a scale change that creates imprecision.

b. Missing Attributes. One or more attributes may be absent altogether in a database.

c. Combined Attributes. Two or more attributes may get combined in an irreversible way. For example, "cost of labor" and "cost of parts" may get combined into "total cost."

d. Missing Data. The value of a given attribute may be missing for some entities but not others. This can be considered a special case of missing attribute by partitioning the set of entities into one consisting of those for which the attribute value is available and one that is not.

e. Classification. Entities may get grouped into classes, and individual attribute values are replaced by class characteristics. For example, instead of recording maximum crusing speed for individual ships, one might record the maximum speed for each of the classes: destroyers, aircraft carriers, etc.

f. Obsolescence. The data that are available may be out of date, e.g., last year's salary, ship position of yesterday, etc.

g. Measurement Error. Random errors are often introduced in measurement and recording.

h. Data Aggregation. Sometimes, the recorded class characteristics are data dependent, for example, the total salary for each department. We shall call such class characteristics, *aggregated data*.

3. A Model for Queires on Imprecise Databases

Consider an idealized world represented by a mapping:

$$E \xrightarrow{F} V$$

where E is a set of entities and V is a space of values. We assume that all queries are expressible in terms of the schema of the idealized world. The actual database, on the other hand, is an instantiation of a *real-world* schema represented by a mapping

$$E \xrightarrow{h} U$$

where U is the space of observed values. In other words, queries concern f but only $\{h(e), e \in E\}$ is known.

Consider an elementary retrieval query of the form:

Find $f^{-1}(A) = \{e \in E : f(e) \in A\}$

for a specified set A in V. The problem of finding $f^{-1}(A)$ can be stated as a problem of hypothesis testing. For each e in E we wish to decide between

$H = f(e) \in A$

and

$H_0 = f(e) \notin A$

and the decision is to be based on our prior knowledge and the observed database $\{h(e), e \in E\}$. In many cases it is reasonable to assume that for a given e, the decision concerning f(e) depends on only h(e) and not $\{h(e'), e' \neq e\}$. The hypothesis testing problem then is one oftesting H against H_0 using the observation h(e).

Suppose that we restrict ourselves to non-randomized decisions. That is, if $h(e_1) = h(e_2)$ the decision for e_1 and e_2 is always the same. Then, any decision rule corresponds to a partition of the space U into sets $\tilde{A}$ and $U-\tilde{A}$, and

we decide for H iff $h(e) \in \tilde{A}$

The set $||f^{-1}(A)|| = \{E : \text{we decide } H \text{ is true}\}$ is expressible as $h^{-1}(\tilde{A})$ and we have the following:

0191-2216/80/0000-0220$00.75 © 1980 IEEE

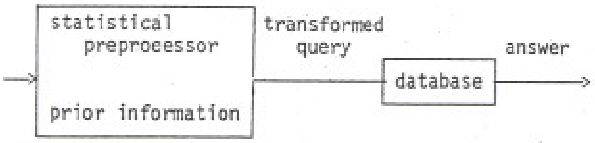
Proposition 3.1 (Separation Principle)

For any nonrandomized decision rule, the approximate answer $||f^{-1}(A)||$ to a to a query $f^{-1}(A)$ is of the form

$$||f^{-1}(A)|| = h^{-1}(\tilde{A})$$

where $\tilde{A}$ depends only on A and the prior distributions.

The separation principle, though little more than observation on our model, has major significances in terms of processing. First, a query on data that we do not have has been transformed into one on data that we do have. Hence, the burden of coping with imprecision is confined to one of query transformation. Second, to perform the transformations requires only prior knowledge and not data. The following processing arrangement is suggested by the separation principle:



The next question is how do we find good decision rules? The answer depends on the specific prior information that we possess. Consider two cases:

Case 1. $h(e)$ is a random variable with a distribution that depends on the value of $f(e)$. We assume that the distribution

$$p(u|v) = \text{prob}(h(e) = u|f(e) = v)$$

is known a priori and does not depend on e. In this case the problem is as follows:

For each e the distribution of $h(e)$ belongs to one of two families:

$$\{p(u|v),\ v \in A\}\ (H)$$

$$\{p(u|v),\ v \notin A\}\ (H_0)$$

We have to decide for each value u which is the case. The situation here is one of testing one composite hypothesis against a composite alternative. A decision rule often used in such a situation is the generalized likelihood ratio test [3].

Define the likelihood ratio by

$$L(u,A) = \frac{\max_{v \in A} p(u|v)}{\max_{v \notin A} p(u|v)}$$

Intuitively, if $L(u,A) >> 1$ then H is more likely, and vice versa. The generalized likelihood ratio test is a one-parameter family of decision rules of the form:

decide H is true iff $L(h(e),A) \geq \alpha$

The parameter α is adjusted according to how one feels about the two types of errors:

miss(decide H_0 when H is true)

false alarm(decide H when H_0 is true)

Increasing α will reduce false alarm at the expense of having more misses.

We shall denote the approximate answer by

$$||f^{-1}(A)||_\alpha = \{e : L(h(e,A) \geq \alpha\}$$

Case 2. We assume that $f(e)$ is a random variable and the distribution

$$p(v|u) = \text{prob}(f(e) = v|h(e) = u)$$

is independent of e and known a priori. In this case a solution to the hypothesis testing problem is the "minimum cost Bayes decision rule" given as follows:

Proposition 3.2

Suppose that the cost for false alarm is α, and for miss $1-\alpha$. Then, the average cost is minimized by the decision rule:

decide H iff $p(A|h(e)) \geq \alpha$

where

$$p(A|u) = \sum_{v \in A} p(v|u)$$

Proof. For a given $h(e)$ one can decide in one of two ways. If we decide for H, the cost is that of false alarm α and the probability of having a false alarm is $1-p(A|h(e))$. Similarly, the weighted cost if we decide for H_0 is $(1-\alpha)p(A|h(e))$. Hence, the decision rule that minimizes average cost is to choose the smaller of the two

$$\alpha[1-p(A|h(e))],\ (1-\alpha)p(A|h(e))$$

or

decide H if $p(A|h(e)) \geq \alpha$

decide H_0 otherwise Q.E.D.

The family of sets

$$||f^{-1}(A)||_\alpha = \{e : p(A|h(e)) \geq \alpha\}$$

decreases with increasing α, and represents a family of approximations to $f^{-1}(A)$. The parameter α is adjusted according to the relative cost that is assigned to "false alarm." Observe that in terms of Lipski's upper and lower bounds, we have

$$||\ \ ||_* = ||\ \ ||_1$$

$$||\ \ ||^* = ||\ \ ||_{0+}$$

In most cases where probabilities are available, these limiting bounds are not useful approximations.

Since the decision rule given by Proposition 3.2 is nonrandomized, the separation principle applies. If we write

$$\tilde{A}_\alpha = \{u \in U : p(A|u \geq \alpha)\}$$

then

$$||f^{-1}(A)||_\alpha = h^{-1}(\tilde{A}_\alpha)$$

and the original query has been modified into a query on the actual database.

Summarizing, for a query of the form

Find $\{e : f(e) \in A\}$

we propose the following as approximate answers:

Case 1: $||f^{-1}(A)||_\alpha = \{e : L(h(e),A) \geq \alpha\}$

Case 2: $||f^{-1}(A)||_\alpha = \{e : p(A|h(e)) \geq \alpha\}$

In each of these cases the separation principle applies and we can write

$$||f^{-1}(A)||_\alpha = h^{-1}(\tilde{A}_\alpha)$$

with

$$\tilde{A}_\alpha = \begin{matrix} \{u = L(u,A) \geq \alpha) \\ \text{or} \\ \{u = p(A|u) \geq \alpha\} \end{matrix}$$

4. An Example

Let E be a collection of ships, each identified by name, and let the idealized database consist of:

$f(e) = (\text{type}(e),\text{speed}(e),\text{current location}(e))$

The space V is defined by

$V_{\text{type,speed}} = \{(\textit{carrier},[20,30]),(\textit{sub},[25,40])\}$

$V_{\text{location}} = \{\textit{Atlantic},\textit{Pacific},\textit{Indian},\textit{Med}\}$

The actual database consists of:

$h(e) = (\text{type}(e), \text{last week's location}(e))$

Suppose that our prior knowledge can be summarized by t he probabilities

p(current loc|LWL) and p(speed|type)

as indicated below

	M	.1	0	.2	.8
current	I	0	.15	.8	.1
	P	.1	.8	0	0
	A	.8	.05	0	.1
		A	P	I	M
			LWL		

Figure 4.1. p(current|LWL)

a in knots	carrier	sub
20	0	0
25	.4	0
30	.1	.2
35	1	.8
40	1	1

Figure 4.2. p(speed < p|type)

Now consider the following query:

"Find all ships in Mediterranean with speed $\geq$ 30 knots."

The probability

$p(\text{speed} \geq 30, \text{loc} = \text{Med}|\text{type},\text{LWL})$

can be easily computed, and we find

type = carrier: p = 0 for all LWL

type = sub:

LWL	A	P	I	M
p	.08	0	.16	.64

Therefore, $\tilde{A}_\alpha$ is given by

$\tilde{A}_\alpha = 0 \qquad \alpha > .64$

$(\text{type=sub},\text{LWL=Med}) .16 < \alpha \leq .64$

$(\textit{type=sub},\text{LWL} \in \{\text{I},\textit{Med}\}) .08 < \alpha \leq .16$

$(\textit{type=sub},\text{lwl} \neq \textit{Pacific}) .0 < \alpha \leq .08$

This has been determined without reference to the actual database. Q.E.D.

Remarks

(a) If we replace $\{h(e),e \in E\}$ by $\{f(e),p(e)\}$ where $f(e) = v(h(e))$ and $p(e) = \Pi(h(e))$, then for $\alpha > 1/2$

$$||f^{-1}(\{v_0\})||_\alpha = \{e : f(e) = v_0 \text{ and } p(e) \geq \alpha\}$$

so that we would no longer need the original data h(), or theprior distributions.

(b) The restriction $\alpha > 1/2$ is intuitively reasonable, since we would not expect preprocessing to work unless the data were reasonably "clean."

5. Combined Queries

Thus far, we have only dealt with atomic queries. The question is: what happens when queries are combined, e.g., under Boolean operations? Can the approximate answer to the combined query be expressed in terms of the approximate answers of its components? This is the same question that was posed in [1] for the upper and lower bounds that he introduced. Our treatment of this topic is not yet complete. Here, we present some results on the two most frequently occurring operations: *conjunction* and *existential quantification*. These results are limited to Case 2 where the a posteriori distribution of f(e) given h(e) is known.

Consider a query of the form: Find $\{e : f(e) \in A \cap B\}$. Since

$$f^{-1}(A \cap B) = f^{-1}(A) \cap f^{-1}(B)$$

we can normally process the true components in the conjunction one at a time. This possibility is extensively exploited in query processing alsorithms, especially where data are dispersed [4]. The question that we shall consider here is whether a conjunctive query remains conjunctive when imprecision is involved.

The specific question is:

$$||f^{-1}(A \cap B)||_\alpha \overset{?}{=} ||f^{-1}(A)||_\alpha \cap ||f^{-1}(B)||_\alpha$$

or equivalently

$$(A \cap B)_\alpha \overset{?}{=} \tilde{A}_\alpha \cap B_\alpha$$

The answer is an immediate no! This may appear to severely limit our ability to decompose conjunctive queries when imprecision is involved. However, the following theorem shows that this need not be the case.

Theorem 5.1. The approximations $||f^{-1}(A)||_\alpha$ defined in Section 3 for Case 2 satisfy the following relationships under intersection:

$$||f^{-1}(A \cap B)||_{\alpha+\beta-1} \supset ||f^{-1}(A)||_\alpha \cap ||f^{-1}(B)||_\beta \supset ||f^{-1}(A \cup B)||_{\max(\alpha,\beta)} \tag{5.1}$$

$$||f^{-1}(A)||_\alpha \cap ||f^{-1}(B)||_\alpha \supset ||f^{-1}(A \cap B)||_\alpha \supset ||f^{-1}(A)||_{\frac{1+\alpha}{2}} \cap ||f^{-1}(B)||_{\frac{1+\alpha}{2}} \tag{5.2}$$

If for every e, prob(f(e) ∈ B) = 0 or 1 then

$$||f^{-1}(A \cap B)||_\alpha = ||f^{-1}(A)||_\alpha \cap ||f^{-1}(B)||_1 \tag{5.3}$$

If A and B are conditionally independent given the observation, i.e., $p(A \cap B|u) = p(A|u)p(B|u)$ for all $u \in U$, then

$$||f^{-1}(A)||_\alpha \cap ||f^{-1}(B)||_\beta \subset ||f^{-1}(A \cap B)||_{\alpha\beta} \tag{5.4}$$

Proof: We begin with the elementary equality

$$\text{prob}(f(e) \in A \cup B) = \text{prob}(f(e) \in A) + \text{prob}(f(e) \in B) - \text{prob}(f(e) \in A \cap B)$$

which reflects the fact that probability is additive for disjoint events. Since any probability is bounded from above by 1, we have

$$1 \geq \text{prob}(f(e) \in A) + \text{prob}(f(e) \in B) - \text{prob}(f(e) \in A \cap B)$$

or

$$\text{prob}(f(e) \in A \cup B) \geq \text{prob}(f(e) \in A) + \text{prob}(f(e) \in B) - 1$$

It follows that

$$e \in ||f^{-1}(A)||_\alpha \cap ||f^{-1}(B)||_\beta$$

$$\Longleftrightarrow \text{prob}(f(e) \in A) \geq \alpha \text{ and } \text{prob}(f(e) \in B) \geq \beta$$

$$\Longrightarrow \text{prob}(f(e) \in A \cap B) \geq \alpha + \beta - 1$$

$$\Longleftrightarrow e \in ||f^{-1}(A \cap B)||_{\alpha+\beta-1}$$

We have proved the left half of (5.1). Taking $\alpha = \beta$ and making a change of parameter, we have also proved the right half pf (5.2).

The right half of (5.1) is proved by observing that since $A \cap B$ is contained in both A and B.

$$\text{prob}(f(e) \in A \cap B) \leq \min(\text{prob}(f(e \in A),\text{prob}(f(e) \in B))$$

Hence, $\text{prob}(f(e) \in A \cap B) \geq \max(\alpha,\beta)$

$$\Longrightarrow \text{prob}(f(e) \in A) \geq \max(\alpha,\beta) \geq \alpha, \text{ and}$$

$$\text{prob}(f(e) \in A) \geq \max(\alpha,\beta) \geq \beta$$

and the right half of (5.1) is proved. The left half of (5.2) follows by setting $\alpha = \beta$.

If prob(f(e) ∈ B) = 0 or 1 for every e then for each e

$$\text{prob}(f(e) \in A \cap B\ h(e)) = 0 \quad \text{if prob}(f(e) \in B) = 0$$

$$= \text{prob}(f(e) \in A\ h(e)) \quad \text{if prob}(f(e) \in B) = 1$$

Hence, (5.3) follows.

Finally, if $p(A \cap B|u) = p(A|u)p(B|u)$ for every $u \in U$, then

$$\text{prob}(f(e) \in A \cap B|h(e)) = \text{prob}(f(e) \in A|h(e))\ \text{prob}(f(e) \in B|h(e))$$

Hence

$$e \in ||f^{-1}(A)||_\alpha \cap ||f^{-1}(B)||_\beta \Longleftrightarrow$$

$$\text{prob}(f(e) \in A|h(e)) \geq \alpha \text{ and } \text{prob}(f(e) \in B|h(e)) > \beta$$

$$\Longrightarrow \text{prob}(f(e) \in A \cap B|h(e)) \geq \alpha\beta$$

$$\Longrightarrow e \in ||f^{-1}(A \cap B)||_{\alpha\beta} \qquad \text{Q.E.D.}$$

Remarks

(a) Suppose that for some α and β

$$(A \cap B)_{a+\beta-1} = (A \cap B)_{\max(\alpha,\beta)}$$

Then equalify obtains in (5.1) and exact decomposition of the intersection obtains. Observe that this condition is verifiable in terms of the prior information alone and does not involve data.

(b) (5.3) allows one to decouple the portion of a conjunctive query that references exact data from that which references imprecise data, thereby limiting the effect of imprecise data on processing.

Example

Consider the example of the last section, and let $\underset{\sim}{A}$ = "*speed* ≤ 30" and B = "*loc* = *Med*." We found that $\tilde{A}_\alpha$ remained constant for $.16 < \alpha \leq .64$. Taking $\alpha = \beta = .64$ in (5.1), we get

$$||f^{-1}(A \cap B)||_{.28} \supset ||f^{-1}(A)||_{.64} \cap ||f^{-1}(B)||_{.64} \supset ||f^{-1}(A \cap B)||_{.64}$$

Since the outer limits are equal, we have equality in this case.

Even when perfect decomposition is not possible, (5.1) and (5.2) allow us to use the answers from decomposed pieces with some measure of confidence. This is especially true when imprecision is not severe and one demands a high degree of confidence in the answer. For such cases, α and β would be taken to be near 1 and $(\alpha+\beta-1)$ does not differ much from $\max(\alpha,\beta)$.

6. Statistical Processing through View Support

In database management systems (particularly relational ones) with facility for supporting views, such facility can be used to support statistical processing. Basically, the idea is to treat the prior information on the distributions as an additional database. A query on the idealized world is then transformed by view-mapping into a query that spans both the real database $\{h(e), e \in E\}$ and the statistical database that contains the prior information. It is important to note the difference between such a procedure and the query transformation procedure suggested by the separation principle. The query transformation involved in view mapping is much simpler, but the resulting query is more complex. In effect, one is using the view-support and query processing facilities that normally exist to achieve the computation needed to transform an ideal-world query into a real-world query.

We shall restrict our attention to the relational system INGRES, but the results are easily adapted to other relational systems of comparable power. Define a statistical sub-database consisting of one or more of relations of the form

distribution(ideal attribute, real attribute, probability)

Each tuple in this relation represents one instance of $p(v|u)$ in the form $(v,u,p(v|u))$. For example, the probabilities of Figure 4.1 would appear as Figure 6.1. Now, suppose that the ideal-world schema consists of one or more relations of the form

rel-ideal(eid,v)

where v stands for one or more attributes and eid is the identifier for e. A tuple from such a relation (if one were available) would be an instance of (e,f(e)). For the example of Section 4, we would have an ideal-world relation.

ship-ideal(shipid,type,speed,current location)

Similarly, a real-world schema would consist of one or more relations of the form

rel-actual(eid,u)

a tuple being an instance of (e,h(e)). Continuing with the example of Section 4, we would have

shipdata(shipid,type,location-last-week)

Let A be a set in V of the form

$A = \{v \in V : v*\alpha\}$

where * is a comparison operator and a is a constant. A query $f^{-1}(A)$ would be expressed in QUEL [5] as

distribution-location

current	last-week	prob
M	M	.8
M	I	.1
M	A	.1
I	I	.8
I	P	.15
I	M	.1
P	P	.8
P	A	.1
A	A	.8
A	M	.1
A	P	.05

Figure 6.1. Distribution of Location

```
range of x is rel-ideal
retrieve into result(x.eid)
where x.v*a
```

Now rel-ideal is not a real relation. But for each α, $||f^{-1}(A)||_\alpha$ is obtained by running the following QUEL query

```
range of x is real-actual
range of y is distribution
retrieve into approximation (x.eid)
where (x.u=y.u)
and sum(y.prob by y.u where y.v*a) ≥ α
```

If the comparison operator * is equality, the query for $||f^{-1}(A)||_\alpha$ can be expressed as

```
range of x is view-α
retrieve into approximation(x.eid)
where x.v=a
```

The view relation view-α is defined by

```
range of x is rel-actual
range of y is distribution
define view view-α(x.eid,y.v)
where (x.u=y.u)
and (y.prob≥α)
```

Consider the example of Section 4 once again. Define a view relation

location-α (shipid,current location)

by

```
range of x is shipdata
range of y is distribution-location
define view location-α(x.shipid,y.current)
where (x.location-last-week = y.last-week)
and (y.prob≥α)
```

The approximation $||$*ships currently in "Med"* $||_\alpha$ would then be represented by executing the view query

```
range of s is location-α
retrieve into approximation(s.shipid)
where s.current = "Med"
```

7. Acknowledgement

Most of this work was done at the Computer Corporation of American, Cambridge, Mass. in connection with the "Multibase" project (DARPA-N00039-80-C-0402). I am grateful to Dr. John M. Smith for suggesting the problem; and to him and Dr. Philip Bernstein for many valuable discussions.

8. References

[1] Lipski, W. "On semantic issues connected with incomplete information databases," ACM Trans. Data-Base Syst. 4(September 1979), pp. 262-296.

[2] Barrow, H. G. and Tenenbaum, J. M. "MSYS a system for reasoning about scenes," AI Center Technical Note 121, SRI International, Menlo Park, CA (1976)

[3] Chernoff, H. and Moses, L. Elementary Decision Theory, John Wiley and Sons, New York, 1959.

[4] Wong, E. "Retrieving dispersed data from SDD-1: A system for distributed databases," Proc. 1977 Berkeley Workshop on Distributed Data Management and Computer Networks, Lawrence Berkeley Laboratory, University of California, May 1977.