An Information-Theoretic Study on Aggregate Responses

Chung-Dak Shum
Richard Muntz

Computer Science Department
University of California, Los Angeles

Abstract

An enumeration of individual objects is not always the best means of information exchange. This paper concerns the problem of providing aggregate responses to database queries. An aggregate response is an expression whose terms are quantified concepts. The tradeoff between the conciseness and preciseness of an aggregate response is studied. Conciseness is measured by the length (the number of terms) of an expression, and preciseness is measured by the entropy or the amount of uncertainty associated with the expression. For a given length, an expression with the minimum amount of entropy is called optimal. Under a one-level taxonomy with the same cardinalities for all leaf concepts, the problem of finding an optimal expression can be solved inexpensively. An efficient heuristic is also proposed for the general one-level taxonomy. For a taxonomy of more than one level, an efficient heuristic is suggested which experiments indicate yields good solutions.

1. Introduction

Conventional responses in database systems, usually given as lists of atomic objects, although sufficient to serve the purpose of conveying information, do not necessarily provide efficient and effective communications between a user and the system. This argument is particularly true when the number of entities or objects which satisfy the query is very large. Consider the personnel database of a large corporation and the query

"Who earns more than 30,000?"

If there is a large number of employees whose salaries are more than 30,000, and if it turns out that all engineers and all managers are in the response, then it seems reasonable for the system to let the user know of the situation. Recently, new notions of answers to queries have been receiving more research interest. For example, in [1], an answer to a query is expressed in terms of both atomic facts and general rules; in [2,3,4], intentional descriptions or concepts are being used as part of an answer.

In [4], expressions for answers are given in terms of concepts and individuals. Exceptions within individual concepts are allowed. Thus responses like

"all engineers except John Smith" or

"all engineers except electrical engineers"

can be expressed where *engineers* and *electrical engineers* are concepts and *John Smith* is an individual. One of the motivations behind such forms of answers is their *conciseness*. Instead of a list of names of, say 99 out of 100 engineers, we can give a short and easily comprehensible answer. However, we see an immediate drawback if, say 40 out of 100 engineers, satisfy the query. No longer are we able to express our answer concisely. A possible "solution" to this problem is to sacrifice *preciseness* for conciseness. In fact, this is not an unreasonable tradeoff. Imagine how often one hears a request like

"Tell me in a few words ... "

Apparently, the questioner is aware of the fact that a precise answer may be a long one and it may not be easy to comprehend, and, therefore, is willing to make the tradeoff. We want to make it clear that by sacrificing preciseness, we do not want to do away with *soundness*. An answer is sound [5] if every listed entity satisfies the query conditions. Thus, we cannot just say "engineers" when obviously some of the engineers do not belong to the answer. On the other hand, we also do not want to list only, say 10 engineers, when we know that 40 of them satisfy the query. That is, we still want our answer to be *complete* [5] in the sense described above. Let us consider the following form of response:

"$\frac{40}{100}$ *engineers*".

The impreciseness of such an answer derives from the fact that we cannot pinpoint the qualified individuals even

though we know who the engineers are. But still it is "sound" and "complete" in a certain sense. In this paper, we are interested in answers of this type and will refer to them as *aggregate responses*.

Aggregate responses are, in fact, very common in statistical databases (databases that are mainly used for statistical analysis). Summary tables, tabular representations of aggregate data, are so important in statistical databases that almost all systems provide some form of limited summary table output formatting capabilities [6]. Our aggregate responses, in essence, correspond to a special type of summary table with percentages as the aggregate output.

Here we will study quantitatively the impreciseness of aggregate responses. We will follow an information-theoretic approach. In Section 2, we introduce the form of expressions which will be used as answers to queries throughout the paper. An entropy measure for the information content of an expression will be defined in Section 3. Then we consider the criteria for measuring the goodness of such expressions and attempt to find efficient algorithms for generating "good" expressions. Section 5 suggests other applications for the entropy measure. We summarize our work in Section 6.

2. Definitions and Notation

We consider a finite domain D of individuals, and *concepts* relative to D. A concept is a unary predicate $C(\cdot)$ defined over D, where C, with possible subscripts, is the label of the concept. For convenience, we will also denote the extension of the predicate $\{x \mid C(x)\}$ by C. The context should suffice to disambiguate. A concept C_1 is said to be subsumed by another concept C_2 if and only if $C_1 \subseteq C_2$. We shall use both set terminology (union, intersection, complementation, set inclusion, difference) and logic terminology (disjunction, conjunction, negation, subsumption) when referring to concepts. Further, we denote the cardinality of a concept C as $|C|$. The *extensional* answer A to a query is simply a subset of D whose elements satisfy the query conditions. The problem is to describe query responses concisely in terms of some predefined concepts.

We are not dealing with an arbitrary collection of concepts; instead, we are interested in a *taxonomy* of concepts.

Definition 2.1 A *taxonomy* is a finite tree whose nodes are labeled by concepts. Any node other than the leaf node has two or more successors. The successor concept of each node is subsumed by its parent concept. The union of all successor concepts of any non-leaf node is equal to the parent concept. A taxonomy is called *strict* if all sibling concepts are mutually exclusive.

Since we will be working mostly with strict taxonomies, the word taxonomy will simply be used to refer to a strict taxonomy unless otherwise stated. An extensional answer A to a query is related to a taxonomy by the following definition.

Definition 2.2 A set of individuals A is *classifiable* by a taxonomy T iff the root concept of T contains A. We also refer to the individuals of A as *qualified individuals*.

Next we look at how to describe an extensional answer A in terms of concepts from a taxonomy T given that A is classifiable by T. To this end, we need the notion of an expression.

Definition 2.3 The alphabet of an expression defined over a taxonomy T is composed of the following:

1. *Concepts*: $C_1, C_2, \cdots$,
   Each concept is a label of a node in T.
2. *Rational Numbers*: $r_1, r_2, \cdots$,
   Each rational number must be between 0 and 1.
3. *Concatenation Operators*: +.

Next we introduce the notion of a term, followed by the syntax of an expression.

Definition 2.4 Let r be rational number and C be a concept. A *term* is a couple $\langle r, C \rangle$ and is simply denoted as rC with no confusion. We sometimes refer to a term as a *quantified concept*.

Definition 2.5 An *expression* over the taxonomy T is defined inductively as follows:

1. A term is an expression.
2. If e_1 and e_2 are expressions, so is e_1+e_2.

Expressions are introduced so that extensional answers can be described in terms of high level concepts, though perhaps imprecisely.

Definition 2.6 Let A be an extensional answer to a query classifiable by a taxonomy T. Then e is an expression for A over T if:

i. For all terms, r_iC_i of e, $\dfrac{|C_i \cap A|}{|C_i|} = r_i$.

ii. $A \subseteq \cup\{C_i\}$ where r_iC_i's are terms of e.

iii. If r_iC_i and $r'_iC'_i$ are terms of e, $C_i \neq C'_i$.

Condition (i) merely gives the meaning of a term in an expression. The first component r_i of a term r_iC_i is the fraction of qualified individuals within the concept C_i. Since we also intend for a term to supply to a user information about the cardinality of its associated concept, r_i is not reduced to lowest common denominator. Condition (ii) ensures that every individual in A is covered by some terms in e. It is in this sense that we consider our expression "complete". Condition (iii) precludes redundant terms from an expression.

Example 1: Consider a taxonomy T of three concepts with $C_0 = \{d_1,d_2,d_3,d_4,d_5\}$ the root concept, and let $C_1 = \{d_1,d_2,d_3\}$ and $C_2 = \{d_4,d_5\}$ be its children. Further, let $\mathbf{A} = \{d_1,d_4,d_5\}$ be the extensional answer. Then it is easy to see that the following are expressions for A over T:

$$"\frac{1}{3}C_1 + \frac{2}{2}C_2"$$
$$"\frac{3}{5}C_0 + \frac{2}{2}C_2"$$
$$"\frac{3}{5}C_0"$$

If a user has full knowledge of T and its associated concepts, the first two expressions then essentially furnish exactly the same amount of information with respect to the extensional answer A. The third expression, on the other hand, tells somewhat less than the previous two. In the next section, we will account for the amount of information associated with such expressions quantitatively.

3. Entropy Preliminaries

When we say $\frac{40}{100}$*engineers* are in the answer set A of a query, there is a certain amount of *uncertainty* associated with the expression. Here we would like to quantify this uncertainty. Informally, in the language of probability theory, the expression can be view as describing a *finite probability space* [1] composed of two mutually exclusive events E_1 and E_2 and their associated probabilities. E_1 is the event that a randomly selected engineer belongs to A and its probability p_1 is $\frac{40}{100}$; whereas, E_2 is the event that the engineer does not belong to A and its probability p_2 is $\frac{60}{100}$. It is well-known [7] that Shannon entropy

$$H(p_1,p_2) = -(p_1 \log p_1 + p_2 \log p_2)$$

is a very suitable measure of the uncertainty involved; the logarithms are taken to an arbitrary but fixed base, and we always take $p \log p = 0$ if $p = 0$. In general, however, we can have n mutually disjoint events.

Definition 3.1 Let S be a finite probability space composed of mutually disjoint events $E_1,E_2,\cdots,E_n$ with probabilities $p_1,p_2,\cdots,p_n$. Then the *Shannon entropy* of the space S is given by

$$H(\mathbf{S}) = H(p_1,p_2,\cdots,p_n) = -\sum_{k=1}^{n} p_k \log p_k$$

First, we review a number of properties this function has and which we might expect of a reasonable measure of uncertainty of a space.

First of all, we see immediately that $H(p_1,p_2,\cdots,p_n) = 0$ if and only if one of the $p_1,p_2,\cdots,p_n$ is one and all the others are zero. But this is just the case where there is no uncertainty as to its outcome. In all other cases the entropy is positive. Furthermore, for fixed n it is obvious that the space with the most uncertainty is the one with equally likely outcomes, that is, $p_k = \frac{1}{n}$ $(k = 1,2,\cdots,n)$, and indeed the entropy assumes its largest value [8]

$$H(p_1,p_2,\cdots p_n) \le \log n = H(\frac{1}{n},\frac{1}{n},\cdots,\frac{1}{n})$$

Next, consider along with space S (events S_i, $1 \le i \le n$), another finite space R with events R_k, $1 \le k \le m$. If it is known that event R_k occurred, then the events S_i of the space S have the new probabilities

$$p_{R_k}(S_i) = \frac{p(S_iR_k)}{p(R_k)} \quad (i = 1,2,\cdots,n)$$

instead of the previous $p(S_i)$. Correspondingly, the previous entropy of the space S

$$H(\mathbf{S}) = -\sum_{i=1}^{n} p(S_i) \log p(S_i)$$

is replaced by the new quantity

$$H_{R_k}(\mathbf{S}) = -\sum_{i=1}^{n} p_{R_k}(S_i) \log p_{R_k}(S_i),$$

which, naturally, we shall regard as the conditional entropy of the space S under the assumption that the event R_k occurred in the space R. A specific value of $H_{R_k}(\mathbf{S})$ is associated with each of the events R_k of space R, so that $H_{R_k}(\mathbf{S})$ can be regarded as a random variable defined over

1. A finite probability space is a set of mutually disjoint events $\{A_i\}$ with probabilities

$$p(A_i)\ (1 \le i \le n,\ p(A_i) \ge 0;\ \sum_{i=1}^{n} p(A_i) = 1)$$

the space R. The expected value of this random variable is the subject of the following definition.

Definition 3.2 Let S and R be two finite probability spaces with events $\{S_i\}$ $(i = 1,2, \cdots, n)$ and $\{R_k\}$ $(k = 1,2, \cdots, m)$, respectively. Then the *conditional entropy* of the space S averaged over the space R is

$$H_{\mathbf{R}}(\mathbf{S}) = \sum_{k=1}^{m} p(R_k) H_{R_k}(\mathbf{S})$$

This quantity indicates the average amount of uncertainty with respect to space S, if it is known which of the events of the space R actually occurred.

Conditional entropy is an important concept in our study. To see this, let us look back at Example 1. First, consider the third expression, $\frac{3}{5} C_0$, for answer A over the taxonomy T. This expression can be translated into a finite probability space S of two mutually exclusive events S_1 and S_2. S_1 is the event that an individual in C_0 belongs to A and its probability of outcome is $\frac{3}{5}$; whereas, S_2 is the event that an individual in C_0 does not belong to A and its probability of outcome is $\frac{2}{5}$. Thus we define the uncertainty of this single-termed expression as the entropy of space S

$$H(\mathbf{S}) = -\left(\frac{3}{5}\log\frac{3}{5} + \frac{2}{5}\log\frac{2}{5}\right) = 0.97$$

Next, we turn to the expression $\frac{1}{3} C_1 + \frac{2}{2} C_2$. In words, this expression can be interpreted as:

i. If an individual is in C_1, then the probability that it belongs to A is $\frac{1}{3}$.

ii. If an individual is in C_2, then the probability that it belongs to A is $\frac{2}{2}$.

Now we can introduced another finite probability space R of two events R_1 and R_2. R_1 is the event that an individual belongs to C_1 and its probability is $\frac{3}{5}$; whereas, R_2 is the event that it belongs to C_2 and its probability is $\frac{2}{5}$. Suppose one has full knowledge of the taxonomy T, its concepts and the individuals associated with each concept. Then the uncertainty of this two-termed expression for the answer A should correspond to the conditional entropy of the space S averaged over the space R

$$H_{\mathbf{R}}(\mathbf{S}) = \frac{3}{5} H\left(\frac{1}{3},\frac{2}{3}\right) + \frac{2}{5} H\left(\frac{2}{2},\frac{0}{2}\right) = 0.55$$

It should not be surprising that the amount of uncertainty in this latter expression is smaller than the previous one. In fact, the expression $\frac{3}{5} C_0$ can be derived from $\frac{1}{3} C_1 + \frac{2}{2} C_2$, together with the full knowledge of each concept. It should be note that, however, in general [8]

$$H_{\mathbf{R}}(\mathbf{S}) \le H(\mathbf{S})$$

What this amounts to is that, on the average, the amount of uncertainty in the space S can either decrease or remain the same, if it is known which event occurred in some other space R. The uncertainty of a situation cannot be increased as a result of obtaining additional information.

Notice that the concepts C_1 and C_2 in the expression above are disjoint and thus the conditional space R can be constructed in a straightforward manner. In general, the concepts associated with an expression for A over T are not necessarily disjoint. For instance, it is quite natural for an expression to have the form

$$\text{"}\frac{40}{100}\ \textit{engineers} + \frac{30}{40}\ \textit{electrical engineers}\text{"},$$

where say, the total number of *engineers* and *electrical engineers* are 100 and 40 respectively. Obviously, *electrical engineers* are *engineers*, and thus, the two concepts are not disjoint. But this expression can, equivalently, be rephrase as

$$\text{"}\frac{30}{40}\ \textit{electrical engineers} + \frac{10}{60}\ \textit{other engineers}\text{"},$$

where *other engineers* are all *engineers* except *electrical*. Now these two concepts are mutually disjoint. We can form the probability space R of two events and evaluate the uncertainty of this expression as the conditional entropy averaged over the space R. More formally, and more generally, we have the following definition:

Definition 3.3 Let $e = r_1 C_1 + \cdots + r_m C_m$ be an expression for an answer A over a taxonomy T and let R be the root concept of T. Define

$$I_j = \{i \mid C_i \subset C_j \text{ and there is no } r_k C_k \text{ in } e \text{ such that } C_i \subset C_k \subset C_j\}$$

$$\hat{C}_j = C_j - \bigcup_{i \in I_j} \{C_i\}$$

$$\hat{r}_j = \frac{r_j |C_j| - \sum_{i \in I_j} r_i |C_i|}{|\hat{C}_j|}$$

for $j = 1,2, \cdots, m$. The *entropy* for e is

$$H(e) = \sum_{j=1}^{m} \frac{|\hat{C}_j|}{|R|} H(\hat{r}_j, 1 - \hat{r}_j)$$

It is not difficult to see that the $\hat{C}_j$'s are disjoint sets of in-

dividuals and their corresponding $\hat{r}_j$'s specify the fractions of qualified individuals within the C_j's. With the entropy of expressions defined, we can now compare two expressions.

Definition 3.4 Two expressions e_1, e_2 for an answer A over a taxonomy T are *equivalent* iff $H(e_1) = H(e_2)$.

Notice that two equivalent expressions may involve different concepts. The first two expressions in Example 1 illustrate such a case.

4. Expression Conciseness

We have defined alternative ways for presenting an answer to a user, not as an exhaustive list of individuals, but rather as an expression of quantified concepts. Such expressions, in general, can no longer be regarded as precise answers; however, it is often possible to express them in a concise manner. Since conciseness is one of our main concerns here, the number of terms appearing in an expression is clearly an important criterion against which to measure how good such an answer is. The simplest expression (one with the least number of terms) for an answer A over a taxonomy T is, of course, a single-termed expression with the root as its only quantified concept. However, the amount the uncertainty introduced in such a single-termed expression is usually very high. Clearly, there is a tradeoff between the length (i.e. number of terms) of an expression and its associated uncertainty. With our entropy measure of expression uncertainty, we can formulate the following problem.

OPTIMAL EXPRESSION

INSTANCE: An extensional answer A to a query classifiable by a taxonomy T and a positive integer K.

PROBLEM: Find an expression e for A over T such that the length of e is no more than K and for any other expression e' for A over T whose length is no more than K, $H(e) \leq H(e')$.

4.1 The Greedy Approach

A naive approach to the OPTIMAL EXPRESSION problem is to form all expressions (for the extensional answer A over the taxonomy T) of length K and identify the one with the minimum entropy. However, such an approach quickly becomes impractical as the taxonomy T and the allowable length K grow. In fact, it is not hard to see that the number of expressions to check in this simple algorithm increases in $O(N^K)$; where N is number of concepts in the taxonomy T.

It may appear that if we do not insist on obtaining the OPTIMAL EXPRESSION, a "greedy" algorithm will probably lead to "good", although perhaps not optimal solutions. Unfortunately, we are able to show that even for a simple one-level taxonomy, the seemingly plausible "greedy" algorithm can result in unbounded relative error. This can be shown by constructing instances in which it behaves arbitrarily badly. Figure 1 shows the algorithm. Let us consider this algorithm applied to the taxonomy T in Figure 2.

Input: A 1-level taxonomy with root R and leaves $\{C_1, \cdots, C_N\}$
An extensional answer A classifiable by T
Allowable length of expression k ($k < N$)

Output: Expression E for A over T with length k

begin
  $E := \frac{|A \cap R|}{|R|} R$;
  while the length of E is less than k **do**
  begin
    add a term $\frac{|A \cap C_i|}{|C_i|} C_i$ to E to form E'
      such that $H(E) - H(E')$ is maximum;
    (**comment:** break ties arbitrarily)
    $E := E'$
  end
end

Fig 1. Greedy Heuristic for OPTIMAL EXPRESSION

The number accompanying each concept in the figure represents the fraction of the individuals within the concept which are in the extensional answer. Suppose the number of allowable terms for the expression is three. Then it can easily be verified that the optimal expression is

$$E_{opt} = \frac{38}{60} C_0 + \frac{1}{10} C_4 + \frac{0}{10} C_5$$

First, we give an intuitive explanation for the expression E_{opt}. It should be obvious that any expression with less than five terms here must include the root as a quantified concept; otherwise, it cannot "completely" describe the extensional answer A. Next, consider the information regarding those concepts which are not explicitly represented in the expression, but which can be derived from the response; the only such information is the fraction of

qualified individuals in the union of excluded concepts. For example, C_1, C_2 and C_3 are such concepts in the above expression and the only information available is that an individual belonging to any one of those concepts has a $\frac{37}{40}$ chance of belonging to A as well. Since the uncertainty associated with this quantity is relatively small, so is the entropy ($H(E_{opt}) = 0.334$).

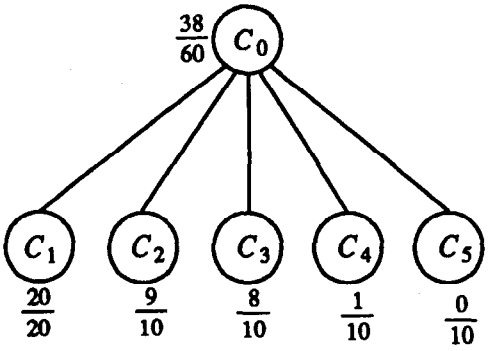

Fig 2. An Example One-Level Taxonomy

In the greedy algorithm, each iteration adds a term to the expression which results in the maximum decrease in entropy. Since all of the individuals in C_1 belong to A, and C_1 also has a relatively large cardinality, intuitively, the term $\frac{20}{20}C_1$ has the most information and, indeed, adding it to the expression results in the maximum decrease in uncertainty. Similarly, the term $\frac{0}{10}C_5$ is included in the next step. Now the resulting expression has the form

$$E_{alg} = \frac{38}{60}C_0 + \frac{20}{20}C_1 + \frac{0}{10}C_5$$

and its entropy $H(E_{alg}) = 0.485$. Not only is E_{alg} not optimal, but by scaling up the cardinalities of the concepts and adjusting the number of qualified individuals within each concept appropriately, the relative error on the entropy can be made arbitrarily large. Figure 3 shows such a construction. By similar intuitive arguments, it is not difficult to see that the optimal expression and the expression resulting from the greedy algorithm will have the following forms

$$E_{opt} = \frac{4m-2}{6m}C_0 + \frac{1}{m}C_4 + \frac{0}{m}C_5$$

$$E_{alg} = \frac{4m-2}{6m}C_0 + \frac{2m}{2m}C_1 + \frac{0}{m}C_5$$

By Definition 3.3, we obtain their entropies as

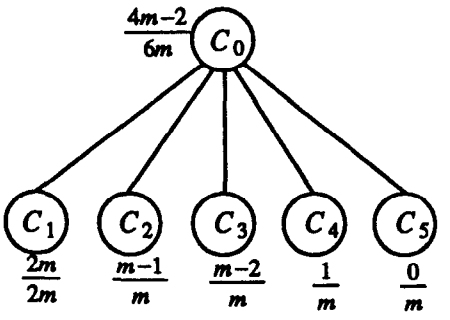

Fig 3. Unbounded Relative Error for Greedy Algorithm

$$H(E_{opt}) = \frac{4}{6}H(\frac{4m-3}{4m}, 1-\frac{4m-3}{4m}) + \frac{1}{6}H(\frac{1}{m}, 1-\frac{1}{m})$$

$$H(E_{alg}) = \frac{3}{6}H(\frac{2m-2}{3m}, 1-\frac{2m-2}{3m})$$

Obviously, as m increases, $H(E_{opt}) \to 0$; whereas, $H(E_{alg}) \to 0.459$. Thus, the relative error can grow unbounded.

4.2 A Restricted Problem

If we consider a one-level taxonomy in which the cardinalities of the all leaf concepts are the same, the OPTIMAL EXPRESSION problem becomes much simpler to solve. Although the problem may now look too restricted to have practical value, there is an important property associated with the solution which, we will show, leads to a useful heuristic for approaching the general problem. In this subsection, we establish this important property and see how a simple algorithm for the restricted OPTIMAL EXPRESSION problem falls out naturally as a result.

Let us begin with a simple case. Consider a one-level taxonomy T with root concept R_0 and leaf concepts $\{C_1, \cdots, C_N\}$ where $|C_1| = \cdots = |C_N|$. For convenience, we also assume, throughout the rest of our discussion, that for any two terms r_iC_i and r_jC_j $(1 \le i,j \le N)$, if $i < j$, $r_i < r_j$. Of course, each $r_i = \frac{|A \cap C_i|}{|C_i|}$ where A is the extensional answer. Now suppose we want an optimal expression for A over T with length $N-1$. Since the number of leaf concepts is N, they cannot all be included in the expression. For "completeness", the root must be included. Now the remaining task involves picking two leaf concepts to be excluded from the expression. We claim that for the expression to be optimal, the two excluded concepts must have the following property.

Lemma 4.1 Let r_iC_i and r_jC_j be the two terms to be excluded from the optimal expression and $i < j$. Then $j = i + 1$; that is, the two excluded terms are consecutive in the ranking by r_i.

Proof: As mentioned before, the information regarding the excluded terms is only an aggregated fraction of qualified individuals. In this case, the quantity is $\frac{r_i + r_j}{2}$ since the cardinalities of C_i and C_j are the same. Define

$$G = \frac{2}{N} H(\frac{r_i + r_j}{2}, 1 - \frac{r_i + r_j}{2}) - \frac{1}{N} H(r_i, 1 - r_i) - \frac{1}{N} H(r_j, 1 - r_j)$$

as the gain in entropy as a result of excluding r_iC_i and r_jC_j. Minimizing G is therefore equivalent to minimizing the response expression.

Assuming that r_iC_i is not in the optimal expression, we want to find r_jC_j. Assume r_j is continuous for the moment, and differentiate G with respect to r_j

$$\frac{dG}{dr_j} = \frac{1}{N} H'(\frac{r_i + r_j}{2}, 1 - \frac{r_i + r_j}{2}) - \frac{1}{N} H'(r_j, 1 - r_j)$$

where $H'(\bullet) = \frac{d\,H(\bullet)}{d\,\bullet}$. Since $H'(\bullet)$ is a decreasing function (Figure 4), $\frac{dG}{dr_j}$ is always positive for $r_i < r_j$. So the closer r_j is to r_i, the smaller is the entropy of the expression and, thus, the lemma follows.

□

With a simple modification, the constraint $|C_i| = |C_j|$ can be relaxed from the proof of Lemma 4.1. Let us restate the Lemma in a slightly more general form which will be useful in establishing the next theorem.

Lemma 4.2 Let r_iC_i and r_jC_j be the two terms excluded from the optimal expression ($|C_i|$ not necessarily equal to $|C_j|$) and $i < j$. If there exists a term r_kC_k such that $i < k \le j$ and $|C_k| \le |C_j|$, then $j = k$.

Proof: Similar to the proof of Lemma 4.1. Refer to [9] for details.

□

We now return to the restriction that $|C_1| = \cdots = |C_N|$. Consider the problem of finding an optimal expression for A over T with length K ($K < N$). Again the root has to be included for "completeness", and the problem amounts to choosing k ($= N - K + 1$) terms to be excluded from the expression. The property for the two excluded terms in Lemma 4.1 is now extended to the case in which k terms are to be excluded.

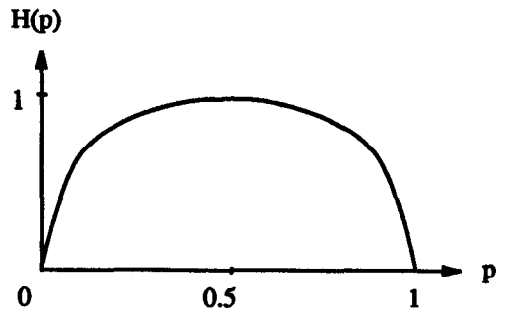

Fig 4a. Entropy Plot

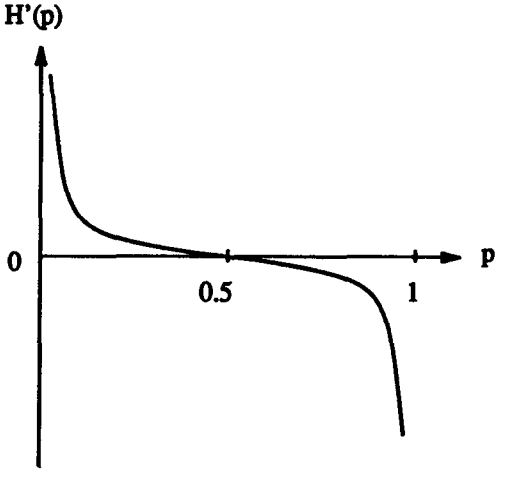

Fig 4b. Entropy Derivative Plot

Theorem 4.1 (below) basically says that the excluded terms $r_{i_j}C_{i_j}$'s have to be consecutive in the ordering by value of r_i. That is, if $\frac{1}{10}C_1$, $\frac{3}{10}C_2$, $\frac{5}{10}C_3$, $\frac{7}{10}C_4$, $\frac{9}{10}C_5$ are terms, and if three terms are to be excluded, then they must be one of the following

$$\left\{\frac{1}{10}C_1, \frac{3}{10}C_2, \frac{5}{10}C_3\right\}, \quad \left\{\frac{3}{10}C_2, \frac{5}{10}C_3, \frac{7}{10}C_4\right\}, \quad \left\{\frac{5}{10}C_3, \frac{7}{10}C_4, \frac{9}{10}C_5\right\}$$

in order for the expression to be optimal. Now it should be obvious that an algorithm for the restricted OPTIMAL EXPRESSION problem only requires checking $N-K+1$ expressions; where N is total number of concepts and K is the allowable length of the expression.

Theorem 4.1 Let $\{ r_{i_1}C_{i_1}, \cdots, r_{i_k}C_{i_k} \}$ be the k terms to be excluded from the optimal expression and for $1 \le p,q \le k$, $i_p < i_q$ if $p < q$. Then $i_{j+1} = i_j + 1$ for $1 \le j \le k-1$. (That is, the excluded terms must be consecutive in the ranking by r_i.)

Proof: Assume the contrary, that is, there exists a term r_xC_x such that $i_p < x < i_q$ and $r_{i_p}C_{i_p}, r_{i_q}C_{i_q}$ are terms excluded from the optimal expression, while r_xC_x is included.

Now consider the aggregate quantity r_y for the set of excluded terms. There are two cases:

(i) Suppose $r_y \le r_x$.

Define another aggregate quantity r'_y for the set of excluded terms minus the term $r_{i_q}C_{i_q}$. We claim that $r'_y < r_y$. The simple proof is omitted.

Now treat $r'_yC'_y$ as a *pseudo* term. By Lemma 4.2, the expression with $\{r'_yC'_y, r_xC_x\}$ excluded has a lower entropy than the one with $\{r'_yC'_y, r_{i_q}C_{i_q}\}$. **Contradiction.**

(ii) The case with $r_y > r_x$ can be proven similarly.

□

Next we give an intuitive explanation for the theorem. Consider concepts of the same cardinality. If the fraction of qualified individuals within a concept is greater than $\frac{1}{2}$, then the more qualified individuals the concept has, the more information it contains. As an example, for 1000 individuals with groups of 100 each, saying that 90 in a particular group belong to the answer set is certainly more informative than saying that 60 in another group belong to the answer. The reverse is true if the fraction of qualified individuals within a concept is less than $\frac{1}{2}$. Thus for an optimal expression, the included concepts must contain the most and/or the least qualified individuals, leaving the excluded concepts as described by Theorem 4.1.

Note that it is this property of the solution which allows us to significantly reduce the search space for the optimal expression. In the next subsection, we adapt this property as a heuristic in the more general case in which the cardinalities of the leaf concepts are not all the same. This then leads to an approach to the general OPTIMAL EXPRESSION problem.

4.3 The Heuristic

We still assume a one-level taxonomy T, But we no longer require the cardinalities of the leaf concepts to be equal. Consider the problem of finding an optimal expression for an extensional answer A over T with length K. Without the cardinality restriction, Theorem 4.1 no longer holds. We can demonstrate this by the example in Figure 5.

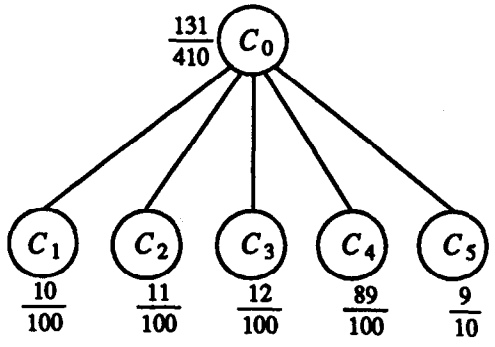

Fig 5. Counter Example for Theorem 4.1

It can easily be verified that $\frac{131}{410}C_0 + \frac{89}{100}C_4$ is an optimal expression of length two. To see this, we first note the following fact. If the cardinalities of two concepts are of the same order, say, 100 and 200, and there is a big difference between the fractions of qualified individuals within each concept, for instance, $\frac{1}{100}$ and $\frac{190}{200}$, then the fraction of qualified individuals within the aggregate of the two concepts $\frac{191}{300}$, represents a significant loss of information. Now the concept C_4, with a relatively large fraction $\frac{89}{100}$ of qualified individuals, if excluded from the response, can only have its information approximated by an aggregated fraction which also involves other concepts such as, C_1, C_2 and/or C_3. These latter concepts contain only a small fraction of qualified individuals. Thus, their aggregation with C_4 results in a relatively large loss of information. Excluding the term $\frac{9}{10}C_5$, on the other hand, is not nearly as bad because its cardinality is much smaller. With $\frac{89}{100}C_4$ in the optimal expression, the excluded terms are not consecutive and, therefore, Theorem 4.1 is not always true. However, in place of Theorem 4.1, we can prove a somewhat similar result.

When the cardinalities of concepts are not all the same, for an expression to be optimal, the excluded terms must satisfy the following condition.

Theorem 4.2 Let r_iC_i and r_jC_j be any two terms excluded from the optimal expression. If there exists a term r_kC_k such that $|C_k| \leq |C_i|$ and $|C_k| \leq |C_j|$, and $i < k < j$, then r_kC_k must also be excluded.

Proof: Similar to Theorem 4.1.

□

Consider a simple example. If $\frac{5}{100}C_1$, $\frac{1}{10}C_2$, $\frac{60}{500}C_3$, $\frac{30}{100}C_4$, $\frac{40}{100}C_5$ are terms, and if $\frac{5}{100}C_1$ and $\frac{30}{100}C_4$ are excluded from an optimal expression, then, according to Theorem 4.2, $\frac{1}{10}C_2$ must also be excluded. Unfortunately, unlike Theorem 4.1, this theorem does not lead to a simple algorithm for obtaining the optimal expression. In fact, under some rare cases, the condition from Theorem 4.2 does not help in reducing the search space for the optimal expression at all. These happen when the expressions for the extensional answer over the taxonomy has the following property: for any three non-rooted terms, r_iC_i, r_jC_j, r_kC_k such that $r_i < r_j < r_k$, either $|C_j| > |C_i|$ or $|C_j| > |C_k|$.

Nevertheless, we claim, on the *average*, the condition from Theorem 4.2 does significantly reduce the search space for the optimal expression. We argue, informally, that if the number of concepts N in a one-level taxonomy is large, the number of expressions satisfying Theorem 4.2 only increases slowly with N. First, we assume that the cardinality of a concept, in general, is not related to the fraction of qualified individuals within that concept, and thus, is not related to our assumed ordering of concepts. Suppose two leaf concepts have to be excluded from an optimal expression. If Theorem 4.2 is to be satisfied, it is very unlikely that the two excluded concepts, say, C_i and C_j ($i < j$), are very far apart; that is, $j-i$ is large. For large N, it is not hard to see that $j-i$ is almost independent of N, and the number of expressions satisfying the constraint increases almost linearly as N. Similar arguments hold for the general case of k excluded concepts. So far this discussion has been concerned with the case in which a few concepts are excluded from the expression. Next, we give intuitive arguments for the case where an optimal expression of a few terms is desired, and thus requiring exclusion of a large number of concepts. Consider picking a concept C_i to be included in the expression from a large number of leaf concepts. In order for Theorem 4.2 to be satisfied, it is conceivable that i should either be close to 1 or N. This remains valid as N increases and thus, the number of satisfying expressions only increases slowly with N. Table 1 gives some idea of how good the heuristic is. We generate taxonomies of N leaf concepts. The cardinalities of the leaf concepts C_i's are selected at random. The third column in the table shows the average number of expressions satisfying Theorem 4.2 when an expression with $N-4$ terms is desired. Similarly, column four shows the case when a six-term expression is desired.

No. of Leaf Concepts (N)	Arbitrarily Choose 5 out of N Leaf Concepts	Using Heuristic	
		Exclude 5 Leaf Concepts from Expression	Include 5 Leaf Concepts in Expression
8	56	18.15	23.02
10	252	46.68	46.68
15	3003	196.45	124.08
20	15504	467.88	212.70
25	53150	865.44	310.41
30	142506	1401.02	411.57
35	324632	2039.81	503.87

Table 1. Performance Exposition for the Heuristic

4.4 The General Problem

The heuristic described for the one-level taxonomy does not immediately extend to the OPTIMAL EXPRESSION problem in general. A simple two-level taxonomy is enough to illustrate the difficulty. Suppose that there are t subtrees under the root of a two-level taxonomy, and the number of allowable terms is k ($t < k$). If we know that the root is not going to be included in the optimal expression, and we are also given the optimal allocation of the number of terms in each subtree, then we can easily invoke our one-level heuristic over each subtree and obtain the optimal solution. Unfortunately, it is not obvious at all how to decide the optimal number of terms to use in each subtree and exhaustively trying all possibilities quickly becomes prohibitive as the complexity grows as $O(t^k)$.

Here we propose an algorithm which is not always optimal, but which avoids the combinatorial explosion problem and leads to reasonably good solutions. The algorithm can be viewed as a postorder traversal of the taxonomy, obtaining expressions for subtrees and merging them as the taxonomy is traversed.

First, we describe the data structures used by the algorithm. For each node N, we construct a table TB_N with l_N entries, where l_N is the number of leaves of the subtree T_N rooted at node N. Each entry $TB_N(k)$ $(k = 1, \cdots, l_N)$ is a tuple $(exp\ ,\ etp)$ such that exp is a k-term expression over T_N and etp is the entropy of exp weighted by the cardinality of N. When the algorithm terminates, a k-term *optimal/near optimal* expression is stored in the table entry $TB_R(k)$ where R is the root of the taxonomy. Thus, the construction of the tables TB_N's constitutes the essence of the algorithm.

For the simple case in which N is a leaf node, TB_N has only one entry $(rN\ ,\ H(r, 1-r)*|N|)$ and r is the fraction of qualified individuals within N. Now suppose N is a non-leaf node and N has p children $S_1, \cdots, S_p$. The table TB_N is constructed through the use the children's tables $TB_{S_1}, \cdots, TB_{S_p}$, as well as the heuristic we developed for the one-level taxonomy in the previous section. More precisely, we utilize the heuristic in the following function

$$(exp\ ,\ etp) = opt\_one\_level\,(T,k)$$

where exp is an optimal k-term expression over the *one-level* taxonomy T and etp is the entropy of exp weighted by the cardinality of the root of T. The table TB_N for each node N, is filled in two phases. A concise description of the algorithm is given, followed by an example.

1. *Initialization*

   Define sub-expressions ξ_i $(i = 1, \cdots, p)$, one for each subtree T_{S_i}.
   For $i = 1, \cdots, p$,
   set $\xi_i = TB_{S_i}(l_{S_i}).exp$.

   Form a one-level taxonomy TX as follows:
   i. Make N the root node of TX.
   ii. Make each concept C in ξ_i a child of N. (In fact, all the leaves of T_N become the leaves of TX)

   For $k = 1, \cdots, \sum_{m=1}^{p} l_{S_m}$
   $TB_N(k) = opt\_one\_level\,(TX,k)$.

2. *Amelioration*

   For each ξ_i $(i = 1, \cdots, p)$ such that $|\xi_i| > 1$,
   $gain_i = TB_{S_i}(|\xi_i|-1).etp - TB_{S_i}(|\xi_i|).etp$.
   Pick j such that $gain_j = \min\{gain_i\}$.
   Set $\xi_j = TB_{S_j}(|\xi_j|-1).exp$.
   Set $\xi = \sum_{m=1}^{p} \xi_i$.

   Form a one-level taxonomy TX as follows:
   i. Create $\hat{N}$ as the root of TX.
   ii. For each concept C in ξ such that $C \not\doteq C'$ for all C' in ξ,
   Obtain $\hat{C}$ and $\hat{r}$ according to Definition 3.3.
   Make $\hat{C}$ a child of $\hat{N}$.

   For each concept $\hat{C}$ in TX,
   Remove the corresponding term rC from ξ.

   Let $l_{\hat{N}}$ be the number of leaf nodes of TX.
   For $k = 1, \cdots, l_{\hat{N}}$
   Set $\xi' = \xi$.
   $(sub\_exp_k\ ,\ sub\_etp_k) = opt\_one\_level\,(TX,k)$
   For each concept C in sub_exp_k,
   Add the term rC to ξ'.
   If $H(\xi')*|N| \le TB_N(|\xi'|).etp$ then
   $TB_N(|\xi'|) = (\xi'\ ,\ H(\xi')*|N|)$

3. Repeat 2. until $|\xi_i| = 1$ for all i $(i = 1, \cdots, p)$.

We illustrate the algorithm by giving an example of one application of the amelioration step. Consider the portion of a taxonomy shown in Figure 6. The shaded nodes correspond to concepts appearing in the current expression ξ.

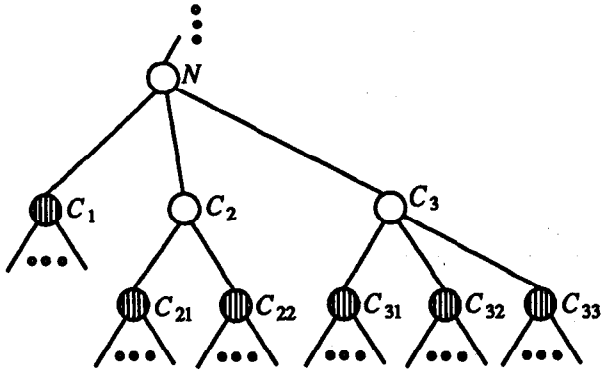

Fig 6. Portion of a Taxonomy

Let the ξ_i expressions at this iteration be

$$\xi_1 = r_1 C_1 + \cdots$$
$$\xi_2 = r_{21} C_{21} + r_{22} C_{22} + \cdots$$
$$\xi_3 = r_{31} C_{31} + r_{32} C_{32} + r_{33} C_{33} + \cdots$$

Note that the "..." in these expressions indicate that other terms corresponding to concepts deeper in the subtrees may be present. Now suppose that expression ξ_3 can be reduced by one term with the minimum increase in entropy and suppose that the new ξ_3, ξ'_3 is:

$$\xi'_3 = r_3 C_3 + r_{32} C_{32} + \cdots$$

The situation is illustrated in Figure 7.

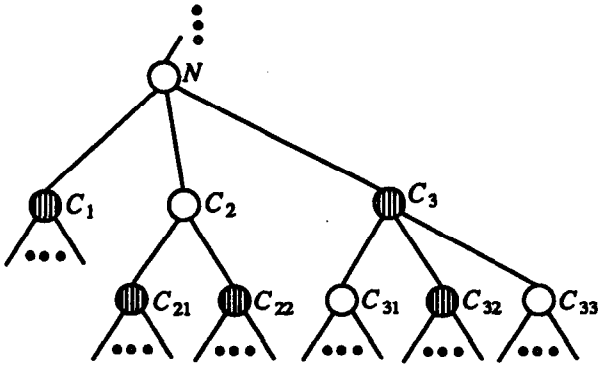

Fig 7. Expression Reduced by one term

Now the one-level taxonomy created in this instance of step 2 will contain concept N and $\hat{C}_i$ for each concept in a cut through the tree involving concepts in T_N which are "as high in the taxonomy as possible". In this case these are $\hat{C}_1, \hat{C}_{21}, \hat{C}_{22}$ and $\hat{C}_3$. The idea is to find an *optimal/near optimal* alternative expression for this portion of the response set corresponding to $\hat{C}_1, \hat{C}_{21}, \hat{C}_{22}, \hat{C}_3$ and replace this portion of ξ if a better expression is found. Intuitively, this allows the algorithm on each iteration to reevaluate the portion of the expression dealing with the higher level concepts.

We have experimented with taxonomies of two/three levels and up to 40 concepts. The cardinality and the number of qualified individuals within each leaf concept were generated at random. Of the 50 examples we tried on the two-level taxonomy, in 43 cases the heuristic generated an optimal expression. For the others, the entropies were no more than four percent higher than the optimal expressions. Another study of 35 cases on a three-level taxonomy, shows a little more than half of the heuristic expressions are optimal and the entropies of the heuristic expressions do not exceed the optimal ones by more than two percent.

5. Other Potential Applications

Aggregate responses, although concise in nature, do provide considerable information to a user. Consider, again, the query

"Who earns more than 30,000?"

and suppose the aggregate response is

$$"\frac{900}{1000} engineer + \frac{50}{50} manager".$$

Since the query has no reference to *job categories*, classifying employees according to their jobs is an extra piece of information. Since *job category* is probably only one of the many possible characterizations of employees, this immediately raises the question of which characterization should be chosen. In fact, under a simple *entity-relationship model* [10], each *attribute* can be used as a characterization for the set of entities or individuals. The issue of *relevance*, an active area of research [11] in Natural Language Processing, has also been receiving much attention with regard to man-machine interfaces for database systems. Here we suggest the use of the entropy measure discussed in this paper as a criterion to select the appropriate characterization. For an extensional answer classifiable by a taxonomy, the lower the entropy is, the better it characterizes the set of qualified individuals and, hence, the more relevant it is to the answer.

In our formulation of the OPTIMAL EXPRSSION problem, entropy is used as a quantitative measure of the preciseness of an aggregate response to a query, but certainly it also has its role in information abstraction. Suppose we have a table describing the male population of different age groups in a city (Table 2a). Now instead of a 10-entry table, we want a summary table of five entries. If we arbitrarily combine pairs of entries from the original table, the result can become quite misleading. Table 2b shows the ratio of male to female population of age group 1-20 is balance; while the truth is that male is dominant in age group 1-10 and female is dominant in age group 11-20. If we evaluate the gain in entropy as a result of combining entries, and choose the one with a minimum gain, we avoid the above discrepancy and obtain a more informative summary table (Table 2c). Once again, we demonstrate the usefulness of our entropy measure.

6. Conclusions

We have considered the problem of providing aggregate responses to database queries. Responses are given in terms of expressions of quantified concepts. The collection of concepts is not arbitrary; instead, it forms a taxonomy. The tradeoff between conciseness and preciseness is studied under a formal information-theoretic framework. Conciseness is measured by the length of an expression, while preciseness is measured by the entropy of the expression. We call an expression of a certain length optimal

Age Group	Population	Male
1-10	100,000	60,000
11-20	120,000	50,000
21-30	150,000	76,000
31-40	170,000	86,000
41-50	160,000	81,000
51-60	120,000	59,000
61-70	70,000	33,000
71-80	30,000	14,000
81-90	5,000	2,300
91-	500	200

Table 2a. Male Population of Different Age Groups.

Age Group	Population	Male
1-20	220,000	110,000
21-40	320,000	162,000
41-60	280,000	140,000
61-80	100,000	47,000
81-	5,500	2,500

Table 2b. Arbitrary Summary of Table 2a.

Age Group	Population	Male
1-10	100,000	60,000
11-20	120,000	50,000
21-50	480,000	243,000
51-60	120,000	59,000
61-	105,500	49,500

Table 2c. Informative Summary of Table 2a.

if its associated entropy is the lowest for that length. Obtaining an optimal expression efficiently turns out to be a challenging task. We show that a seemingly plausible "greedy" algorithm can have unbounded relative error. Under a one-level taxonomy with the same cardinalities for all leaf concepts, the problem can be solved efficiently. An efficient heuristic is also available for the general one-level taxonomy. We also suggest an algorithm for the general OPTIMAL EXPRESSION. Although it is does not always result in an optimal expression, it avoids the combinatorial explosion problem and appears to lead to reasonable solutions.

Acknowledgement

This research was carried out as part of the Tangram project at UCLA and was partially supported by the Defense Advanced Projects Agency under contract No. F29601-87-K-0072.

References

[1] Imielinski, T., "Intelligent Query Answering in Rule Based Systems", *J. Logic Programming*, Vol.4, No.3, September 1987.

[2] Porto, A., "Semantic Unification for Knowledge Base Deduction", *Foundations of Deductive Databases and Logic Programming*, Minker, J.(ed.), August 1986.

[3] Corella, F., "Semantic Retrieval and Levels of Abstraction", *Expert Database Systems*, Kerschberg, L. (ed.), Benjamin Cummings, New York, 1985.

[4] Shum, C.D., Muntz, R., "Implicit Representation of Extensional Answers", *Proc. on 2nd International Conference on Expert Database Systems*, 1988.

[5] Reiter, R., "Towards a Logical Reconstruction of Relational Database Theory", *On Conceptual Modelling: Perspectives from Artifical Intelligence, Data-Base, and Programming Languages*, Brodie, M., Mylopoulos, J., and Schmidt, J. (eds.), Springer-Verlag, New York, 1984.

[6] Ozsoyoglu, G., Ozsoyoglu, Z.M., "Statistical Database Query Languages", *IEEE Tran. Software Engineering*, Vol. SE-11, No. 10, October 1985.

[7] Shannon, C.E., "The Mathematical Theory of Communication", *Bell System Technical Journal*, Vol. 27, 1948.

[8] Khinchin, A.I., *Mathematical Foundations of Information Theory*, Dover, New York, 1957.

[9] Shum, C.D., Muntz, R., "Aggregate Responses", *Technical Report*, Computer Science Department, University of California, Los Angeles, March, 1988.

[10] Chen, P.P-S., "The Entity-Relationship Model - Toward a Unified View of Data", *ACM Transactions on Database Systems*, Vol.1, No.1, March 1976.

[11] Webber, B.L., "Questions, Answer and Responses: Interacting with Knowledge-Base Systems", *On Knowledge Base Management Systems*, Brodie, M. and Mylopoulos, J. (eds.), Springer-Verlag, 1986.