Quantifiers for the Hybrid Use of Data and Knowledge

María de los Ángeles Alonso Lavernia[1], Argelio Víctor de la Cruz Rivera[1], Marco Antonio Pérez Bustos[1]

[1]Center of Research in Technologies of Information and Systems (CITIS)
Autonomous University of the State of Hidalgo (UAEH)
Abasolo 600 Colonia Centro Pachuca, Hidalgo, México CP 42000

Phone: 01-771-72000 Ext. 6732
Fax: 01-771-72109
marial@uaeh.reduaeh.mx
argelioc@uaeh.reduaeh.mx

Abstract. The theoretical results of the development of diverse structures for the Knowledge Representation are presented which allows the development of hybrid intelligent systems based on data and knowledge. Two parameterized quantifiers are defined to generalize the classic quantifiers from the calculation of predicates. These generalizations facilitate the interrogation of a database in a way that answers are associated with a value of certainty which permit to quantify the grade of fulfilment of the condition that it searches within data.
The communication between the Database and a Knowledge Base will be defined through another structure denominated Quantifier Variable, which will establish the appropriate schema of calculus, the relation of search and diverse characteristics of the connection with the database that will be explored.

1. Introduction

The identification of objects, phenomenon or "entities" characterizes the man's activity in his constant desire for increasing his knowledge or to solve the problems that he faces daily. From this observation, he may obtain the necessary information that facilitates him to achieve his objectives.

This situation called "Observational Problem" states a great question: ***What useful knowledge is contained in the data?*** [1].

On the other hand, it is very common in the real life the appearance or presence of problems where the human experience plays a basic role [5], [10]. The medical diagnosis, the prospecting of mineral locations, the taking of managerial decisions and many others are typical examples of these problems.

Then, there are some problems which basic weight is found in these experiences or human knowledge and others where the fundamental role are data obtained of its study. Actually, neither kind of information is negligible. For example, it is known that the doctor sometimes may offer his diagnosis remembering a consulted case, and in other times, he use his own experience. Although, the best solution would be that one that uses both sources of information: ***Data and Knowledge***.

There are some reported works in the literature that search to unify both sources of information. For example, Matt Ginsberg proposes the creation of deductive databases. He uses knowledge representations in the database to extract specific information. It tries to obtain the same results that it would be obtained in a traditional consultation to the database but with the use of knowledge representations for the obtaining of the information [4].

Another related development outlines a unified methodology that represents the data, the information and the knowledge in a homogeneous way, as well as the relationships among them. This methodology builds a maintenance mechanism inside the design, proposing a new structure for the implementation of Knowledge Base which contains data, information and knowledge [3].

The goal in the development of any application of Intelligent database (IBD) is first to understand the processes that generates the information and then to use this information to control or to exploit this processes. An IBD organizes and transmits information, but the application BDI is what interprets the information inside the context of a significant task. Parsaye provides an IBD architecture of three levels: users' interface, high level tools and a database machine [7], [9].

In the current work a theory is developed looking for solutions to problems with the mentioned features. This work offers computational tools for information coming from both sources.

The general idea consists in the construction of structures that allow interrogating a Database (DB) from a Knowledge Base (KB). The KB has the function of controlling the execution of the system. The KB initiates the consulting to DB and this searches for the required condition. The answers of the DB is assigned to the structures of the KB which will be used for the operation of the system, as previously had been defined in the KB.

These structures have been denominated parameterized quantifiers and although they are inspired by the quantifiers of the predicate calculation, they seek to generalize these concepts to be used in the solution of real problems.

With the purpose to establish their differences, we will begin with a brief sketch about the quantifiers of the predicate calculation and then we present the results obtained in the generalization of these.

2. Calculation of Predicates by Means of Quantifiers

In the calculation of predicates, the quantifiers indicate the frequency with which a certain sentence is true. The universal quantifier is used to indicate that a sentence is true if this has been fulfilled to all cases that it is been analyzed, while the existential quantifier indicates that a sentence is true if this has been fulfilled in at least one of all analyzed cases [6], [8].

They are defined in the following way:

Let A be an expression, and x a variable. If we want to indicate that A is true for all possible values of x, we write ? x A, and if we want to indicate that A is true in at least one value of x, we write ? x A.

Thus, ? **x** is called ***universal quantifier*** and ? **x** ***existential quantifier***, A is called scope of the quantifier and we could say that the variable x is tied by the quantifier. The symbol ? is read "***for everything***" and ? expresses "***it exists***."

In general, a quantified expression divides the universe of values of the variable in two groups, one formed by those elements that satisfy the sentence and another one for those that don't satisfy it.

From the practical point of view, these definitions have a restricted use, because of the results are so extreme. For example, if we have a DB of patients with a certain illness and we apply any of these quantifiers to the data. It is only possible to obtain properties that fulfill all registers or to evaluate if a given behavior exists in it. But it won't count how many registers fulfill the condition exactly.

In the following section we will introduce two quantifiers that allow to extend these possibilities.

3. Parameterized Quantifiers

The proposed model generalizes the classic quantifier concepts because it searches for the fulfillment of a sentence in a DB and as result, it offers values of truth that they don't necessarily have to be true or false, and that we will name ***Parameterized Quantifiers*** (**CP**).

We may state that a CP constitutes a Knowledge Representation Form (KRF) in which it can be expressed in different types of queries to be executed toward a DB. The type and features of them are defined by means of the associated parameters. The answer indicates the security or certainty values on the fulfillment of queries. Figure 1 illustrates this process.

The CP is defined as a part of the KB and the process consists on the execution of queries to in DB to act in correspondence with the received answers derived of the interrogation.

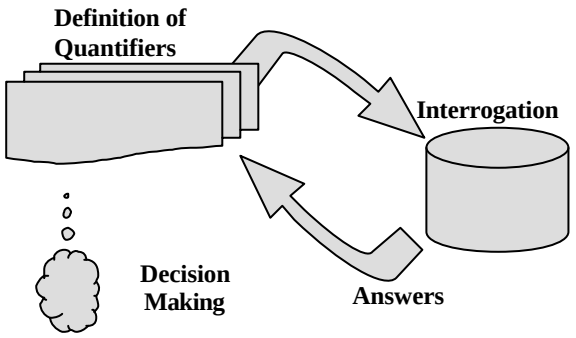


Fig. 1. Operation of Parameterized Quantifiers

The Parameterized Quantifiers have been classified in two types of quantifier that are:

- Existential Parameterized Quantifier
- Universal Parameterized Quantifier

Both represent expressions that allow to design how interrogate the DB, but they differ in that the second one presents an additional relation that determine the set of elements within the specified universe.

3.1. Existential Parameterized Quantifier

Definition 1. – Let be ***Existential Parameterized Quantifiers*** (EPQ) an expression in the way: ? ^{CK} ?**(x)**, where ? (x) represents a relationship between variables or fields of a DB that it should fulfil and the parameters C and K establish the quantity of registers that should fulfill this relationship.

The expression ? ^{CK} ? **(x)** is read: "exists a quantity of objects or registers of the DB, determined by C and K that fulfil the relationship defined in ?(x)".

In the next section, it analyzes how the relationships ?(x) and the parameters C and K are defined.

3.1.1. Definition of the Relationship ?(x) of EPQ

The relationship ?(x) as a part of the definition of EPQ represents a sentence that expresses certain requirements among variables (fields or attributes) of a certain register of the database.

Definition 2. - A relationship ?(x) among fields of a DB for a register k is defined as an expression in the way:

$$?(x) = ?_1 (\& v |) ?_2 (\& v |) ?_3 (\& v |) \ldots (\& v |) ?_r \text{ where:}$$

$?_i = E_1 ? E_2$, i? {1,…, r} r is the number of components of the relationship.

E_i ? $\{V_j(x_k), C_j\}$, $V_j(x_k)$ represents the value of the variable j (field) of the DB for the register k, k ? {1,..., n} n is the number of registers. And C_j is a constant related with the type of data.

? ? {< , > , = , <= , >= , <>}

&, v, | represent the conjunction, inclusive disjunction and excluding disjunction connectives respectively that define the connection among the components of the relationship.

Examples:

1. ?**(x) = Temperature < 25°C.** In this case the relationship is formed by a single component (r = 1) where; E_1 = "Temperature" represents a field of the DB, ? = "<" (the "less than" relation), and E_2 = "25°C" is a constant admitted for the field.

2. ?**(x) = (Quantity of epileptic crisis in the 1st year of illness) >= (Quantity of epileptic crisis in the 2nd year of illness).** In this case, it sates a relationship also constituted by a single component but between two fields of the DB.

3. ?**(x) = (Hair Color = black) v (Hair Color = yellow).** Here is a relationship formed by two components (r = 2) where the demanded fulfillment is of at least one of these because the inclusive disjunction connective: "v", is used.

4. ?**(x) = (Temperature > 38°C) & (Age < 5 years) & (Hemoglobin < 10).** This case defines a conjunctive relationship, which obliges the fulfillment of the three components.

3.1.2. Parameters C and K of the EPQ

The parameters of the EPQ allow us to establish the quantities of registers that should accomplish the relationship defined in ?(x), so that the general condition of the quantifier could be fulfilled.

The parameter K defines a threshold that indicates the quantity of registers that should fulfils the relationship defined in ?(x) for the fulfillment of the quantifier. This can be expressed as a number or as a percentage in the following way:

K ? {n, p} where n is a positive integer number or zero and p is a perceptual value.

The parameter "C" expresses the type of comparison to be considered in connection with the threshold settled in K. This is defined as an element of the group {–, +, =, –=, + =, –+}.

Examples:

1. In the expression ? $^{=100\%}$? (x), C is considered as equal "=" and K = 100% which expresses that the quantifier will have a value of truth if all the objects or registers satisfy the relationship defined in ?(x). One can observe that this definition is equivalent with that of the universal quantifier in the calculation of predicates, which would be a particular case of the EPQ.
2. If we want to represent the following expression: "As minimum 60 objects satisfy a condition ?(x)"; C should take a value of "+ =" and K of 60 to form the expression: ? $^{+=60}$?(x).
3. "The situation ?(x) is very frequent and can be express as: ? $^{+70\%}$?(x), which indicates that it is defined the concept of frequent as a quantity bigger than 70% of the objects.

3.2. Universal Parameterized Quantifier

Definition 3. - It calls ***Universal Parameterized Quantifier*** (UPQ) to an expression in the way: ? ^{CK}?**(x):**?**(x)**, where ?(x) and ?(x) represent relationships between variables or field of a DB and the parameters C and K establish the quantity of regis ters that should fulfill both relationships.

The expression ? ^{CK}?**(x):** ?**(x)** is read: "All objects that fulfil ?(x) also should fulfil ?(x) in a quantity defined by C and K".

The relationships ?(x) and ?(x), that compose the UPQ, constitute similar expressions of the definition 2. The same as the parameters C and K whose meanings are identical to the one explained for the EPQ in the epigraph 3.1.2. Let us see some examples to understand the definition.

Examples:

1. To represent: “of the objects that fulfil a relationship ?(x), most of them have the property ?(x)”. To this condition, the UPQ can use that expression, ? $^{+50\%}$? **(x):** ? **(x)**.

2. The expression ? $^{-5}$? **(x):** ? **(x)** can be interpreted as: “among the objects that fulfil ?(x) is strange the presence of the property ?(x).”

3. “Most of the objects that satisfy ?(x) don't satisfy ?(x)”. It can be expressed as: ? $^{-50\%}$? **(x):** ? **(x)**.

3.3. Treatment of Uncertainty in the Parameterized Quantifiers

Being awared of the definitions of EPQ and UPQ, it can be observed that the obtained answer of the evaluation of the expressions is a value of certainty that can only be true or false. However, it is clear that in real problems is not always possible to talk of the absolute fulfillment of a situation; therefore this represents a certain restriction for the representation of a real situation using the quantifiers in the way that they have been defined in their classic form.

To overcome this limitation, it has been introduced the calculation of uncertainty as a part of the process of evaluation of the quantifiers in a way that can express situations as follows:

- Enough security exists on the fulfillment of a given expression.
- It has little security on the accomplishment of a situation.
- It isn’t known if certain relationship is fulfilled or not.

The value of security associated with a quantifier will be considered as a numeric value within of interval [-1,1] where 1 indicates absolute truth (true), -1 absolute falsehood (false), 0 total ignorance (it is not known) and the rest of the values are different gradations of the belief about a certain fact. Thus, the value 0.9 indicate that a lot of security exists to establish the fulfillment of the analyzed situation, the value -0.3, on the contrary, indicates that the situation is not fulfilled, but this may be affirmed with little confidence or certainty [2].

Definition 4. - An existential or universal parameterized quantifier is inexact if each of them might return a value of certainty in the interval [-1, 1] as a result of its evaluative process, as well as the extreme values: true “1” and false “-1”.

This definition takes us to establish different schemes of calculation from original definitions of EPQ and UPQ, which will be conditioned by the type of defined comparison through the parameter C, as it is presented in the following cases:

Case 1: If C ? {+ =, +} then the parameter K breaks down into other three parameters k_1, k_2, k_3, in such a way that, k_1 represents the cutting between positive and negative certainty, k_2 is the value beyond which the expression is fulfilled with absolute certainty (true) and k_3 is the value below which fail, to fulfill with absolute certainty (false). See Case 1 of Figure 2.

Case 2: If C ? {– =,–} the parameter K also breaks down in the three parameters k_1, k_2, k_3, where k_1 maintain the same previous definition, but in this case k_2 will take a value below which this the expression is fulfilled with complete security and

the value of k_3 represents the value above which the expression is unfulfilled. See Case 2 of Figure 2.

Case 3: If C ? {=} then the parameter K breaks down in three parameters k_1, k_2, F where the $[k_1, k_2]$ represents the interval of values where the certainty is positive and F ? [0,1] defines the fraction of the interval $[k_1, k_2]$ with maximum positive weight (true). See Case 3 of Figure 2.

Case 4: If C ? {–+} the parameter K breaks down in the same way that in the case above but the interval [k1, k2] represents the values where the certainty is negative and F is therefore, the fraction of this interval with extreme negative weight (false). See Case 4 of Figure 2.

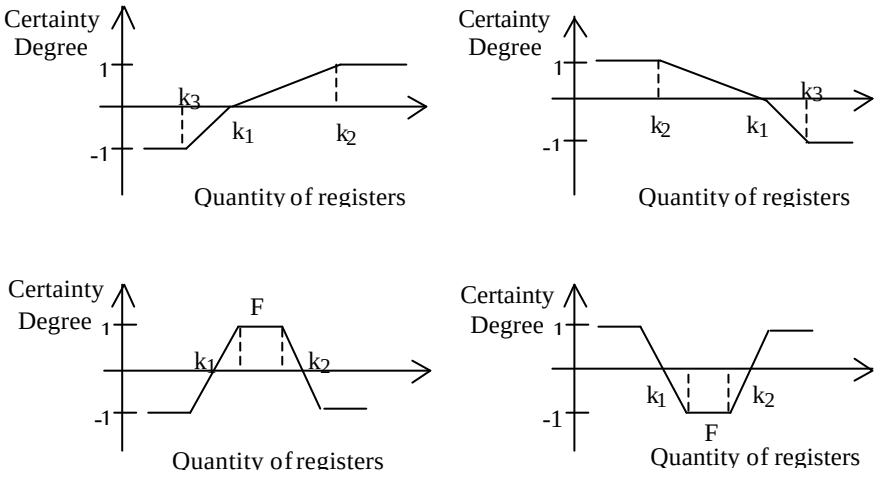


Fig. 2. Schemes of calculation of uncertainty values

As it can be observed, in Figures 2 the abscissas axis represents the quantity of registers that satisfy the relationship ? (x) for the EPQ or ?(x):?(x) for the UPQ, while the ordinates axis represents the values of certainty associated to the quantities of registers that fulfill the relationship above.

Example.- The expression $?^{+70\%, 80\%, 65\%}$?(x), expresses that most of the objects fulfill the relationship ?(x) as follows: if 70% or more fulfill ?(x), then this expression is fulfilled, but the total security is obtained if the quantity is the same or it surpasses 80%, while smaller values than 70% indicates that the expression fails to fulfill, but the absolute security of this one is reached for smaller or same values than 65%. As it can be observed this example corresponds to case 1, which values are k_1 = 70%, k_2 = 80% and k_3 = 65% (Figure 2).

4. Knowledge Base-Database Interface

In order to complete the communication process between a KB and a DB, a new structure of knowledge representation named “Quantifier Variable” is defined.

Definition 5.- Let be ***Quantifier Variable*** V_j a structure that defines the way in which a DB will be interrogated and has the following components:

$$\mathbf{V_j = < Q_j, P_j, CDB_j >} \text{ where:}$$

$\mathbf{Q_j}$ represents a EPQ or UPQ,

$\mathbf{P_j}$ is the associated proposition in the analysis. It is associated to the variable V_j and will take a value of certainty obtained by the interrogation process,

$\mathbf{CDB_j}$ defines the features of the connection with a DB.

The components of the variable are described as:

(1) $\mathbf{Q_j}$ **Definition of the Quantifier**: It describes the features of quantifier to be used for the interrogation of the DB. Their parameters are:

- **Type**: It indicates the type of quantifier, Existential or Universal.
- **Relations**: It expresses the relationship to be interrogated ($?(x)$, $?(x)$).
- **Parameters**: It establishes the comparison type and the exact quantity of registers that the relationship should fulfill (parameters C and K).

(2) $\mathbf{P_j}$ **Associated Proposition**: It is a concept of the KB that takes positive or negative values of certainty in correspondence with the fulfillment or not of the defined quantifier [2]. Regularly, this proposition is the one that activates the interrogation process to the DB.

(3) $\mathbf{CDB_j}$ **Connection with the DB**: It establishes the parameters for the connection with the DB, in terms of Charts, Registers and Fields.

The definition of this variable allows to establish the conditions to interrogate a BD through the parameters that compose their definition. The values of these parameters are very linked to the application that is developed. For example, when we are referring to the fact that most of the registers complete a given relationship, this could be considered indistinctly as 70%, 85% or 90% of the registers of the BD. Then, it could allow the experts to define the criteria and their representation in the quantifier variable.

4.1 Operation of the Quantifier Variable

The quantifier variable constitutes the structure that concentrates the information and controls the whole execution process and communication between the BC and the BD.

The process is activated when the mechanism of inference of the system requires the value of certainty of the central associated proposition to the quantifier variable. Precisely, in that moment, the evaluative process will be activated.

The analysis begins with the interrogation of the BD with the condition defined in the quantifier, taking into account for this the quantity of registers that should accomplish this condition.

Once concluded this process, the evaluation of the proposition is executed using the suitable scheme. This is evaluated with a value of certainty that indicates the degree of fulfillment of the condition according to the data stored in the BD.

The obtained value of this step is returned to the inference machine as a result of the whole process.

5. Example of Use

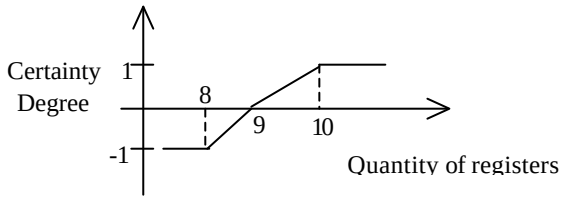
Among the applications that have been developed using the proposed theory there is a system to detect Cervic Uterine Cancer. This system has a Knowledge Base which is designed to execute diverse actions using a database of patients that is being modernizing constantly. One of the considered approaches is to detect possible anomalies in the detected cases taking into account the region or county.

There is a criteria obtained from the experts that mark a threshold of 9% to decide the period of time that should be considered to execute a new journey of prevention and detection.

In the Knowledge Base, the following facts have been considered:

1. 9% or more than the patients present positive diagnosis to cancer.
2. It is recommended to carry out the journey in 6 months.
3. It is recommended to carry out the journey in 18 months.

And according to fact 1 it can be associated a calculation scheme like the one that is shown in Figure 3 that belongs together with Case1 in Figure 2.



However, it is clear that values around 9% they are doubtful.

To represent this situation there has been used a quantifier of the inexact existential type, using the following parameters:

C = bigger or same (+ =)

K1 = 9%, K2 = 10% and K3 = 7%

?(x) = (result = light Cancer) V (result = Invading Cancer) V
(result = Grave Displesia) V (result = Malign)

Being the final expression as: $?^{+9\%,\ 10\%,\ 8\%}$? (x)

Taking the fact number 1 as an associated central proposition to this quantifier, it will happen that when executing the Knowledge Base and to require the value of certainty associated to her, the processing of the variable quantifier will be initialised, which leads to the interrogation of the DB according to the expression ? (x). And according

to their execution, the proposition 1 will be evaluated. This value will be used later on by the knowledge base in its inference to help to the appropriate decision making.

As a result, this system is capable to monitor the information of the database to offer warnings about the presence of anomalies in the stored data.

6. Final Considerations

The exposed knowledge representations in the present work don't constitute an isolated development, since these are part of a knowledge programming language or environment called "HAries", which bases his operation on a group of structures or forms for the knowledge representation and his processing.

The incorporation of this theory by means of the generalization of quantifiers facilitates the development of hybrid intelligent systems, which could mix knowledge coming from human experts and those coming from the analysis of BD related with the problem, which is focused to improve the effectiveness of the applications developed on this base.

The practical use of the developed theory is very wide. For instance, the analysis of student's performance on a given subject, or the decision making to apply cytology tests in a rural area where its previously know that a great amount of women results positive on this tests, or the decision of perforating an area when having evidence of the presence of a petroleum location and many other cases in which enough human experience and collected data of the study for this phenomenon exists.

References

1. Berthold M., Hand D.J.: Intelligent Data Analysis. Springer-Verlag. Berlin Heidelberg New York (1999)
2. De la Cruz A. V.: Fundamentos y Práctica de la Construcción de Sistemas Expertos. Editorial Academia, La Habana, Cuba. (1993)
3. Debenham J.: Knowledge Engineering: Unifying Knowledge Base and Database Design. Springer-Verlag, Berlin Heidelberg, Germany (1998)
4. Gingsberg M. "Essentials of Artificial Intelligence". Morgan Kaufmann Publishers, Inc., San Francisco, CA., U.S.A. (1993)
5. Lakemeyer G., Nebel B.: Foundations of Knowledge Representation and Reasoning. Springer-Verlag, Berlin Heidelberg, Germany (1994)
6. Nilson N. J.: Artificial Intelligence: A New Synthesis. Morgan Kaufmann Publishers, Inc., San Francisco, CA., U.S.A. 1998.
7. Parsaye K., Chignell M.: Intelligent Database Tools and Applications. John Wiley & Sons, inc., U.S.A. (1993)
8. Richmond H.T.: Symbolic Logic, An Introduction. The Macmillan Company/Collier-Macmillan Limited, London.
9. Simon Alan R.: Strategic Database Technology: Management for the year 2000". Morgan Kaufmann Publishers, Inc., San Francisco, CA., U.S.A. 1995.
10. Wooldridge M. J., Veloso M.: Artificial Intelligence Today. Springer-Verlag, Berlin Heidelberg, Germany (1999)