

For Fuzzy Classification of Databases with FCQL

Seyfali Mahini*

Department of Information Technology, Islamic Azad University, Khoy Branch, Khoy, Iran

*Corresponding author: Seyfali Mahini, Department of Information Technology, Islamic Azad University, Khoy Branch, Khoy, Iran, E-mail: my1341post@yahoo.com

Received date: January 30, 2022, Manuscript No. IPMCR-22-12467; **Editor assigned date:** February 01, 2022, PreQC No. IPMCR-22-12467 (PQ); **Reviewed date:** February 15, 2022, QC No. IPMCR-22-12467; **Revised date:** February 20, 2022, Manuscript No. IPMCR-22-12467 (R); **Published date:** February 27, 2022, DOI: 10.36648/2349-3917.10.2.133

Citation: Mahini S (2022) For Fuzzy Classification of Databases with FCQL. Am J Compt Sci Inform Technol Vol.10 No.2: 133

Description

Business information systems have extensive databases that are mainly managed in relational databases. What is often missing are automated procedures to analyze these inventories without major restructuring. Therefore, we extend the relational database schema with a context model for fuzzy classification. Based on this, we develop the fuzzy classification query language, FCQL, which allows fuzzy queries to the extended database schema using linguistic variables and converts them into SQL calls to the database. With this we give the user a data mining tool so that he can start extended queries on his databases based on a predefined fuzzy classification and obtain an improved basis for decision-making

Information Systems

On the way to the information society, a lack of data has turned into an overabundance (information overload). Therefore, companies and organizations are interested in tools for data analysis in order to continue to have a basis for business decisions. Of particular interest is the process of Knowledge Discovery in Databases (KDD), which extracts valuable information from sometimes extensive databases. The primary goal of a KDD process is to reduce the complexity of the data or recognizing patterns in large databases. Classic methods such as cluster analysis or regression analysis are mostly based on statistical methods. They assume that the amounts of data or databases contain numerical values or contain sharp data values. As soon as the data itself or the classes that are defined in the data analysis and to which the data is then assigned to reduce complexity are no longer clearly defined, many conventional data analysis methods fail.

Relational Database

In companies, the databases contain not only numerical data but also non-numerical information. Database query languages require queries to be formulated with the same level of detail and precision as the data is stored in the database. Relational query languages, such as SQL, do not permit imprecisely formulated or fuzzy queries.

illustrates a fuzzy query that uses the vague term 'unacceptable' This small visual example illustrates the need to

be able to operate with vague and imprecise queries in the case of large databases. Even in the case of fuzzy classification, there are many practical examples from everyday business:

Customer Relationship Management: Recently, the management of customer relationships and processes has gained in importance. In addition, the customers are happy to be based on certain characteristics resp. of purchasing behavior divided into classes or customer segments. A common division is into A, B and C customers, depending on the customer's purchasing power. In most cases, this class affiliation is strictly defined, i.e. each customer belongs to exactly one class. If the development potential is to be taken into account in addition to the completed transactions, an individual customer can no longer be clearly assigned to a customer segment. Analyzing the possibly extensive customer base and creating fuzzy customer segments is a must. This is the only way that suitable marketing, sales and after-sales service tasks can be carried out in a targeted and cost-effective manner.

Checking creditworthiness and risk: Insurance companies and banks divide their customers into various classes based on risk considerations. Age, purchasing power and other characteristics must be checked for a loan application. Instead of sharp creditworthiness or risk classes, fuzzy classes that work with the help of a membership function can be interesting. Based on research, it has been found that traditional credit checks and clearly different risks can result in the same overall rating for the customer. Conversely, it is also possible that different overall ratings can arise if the customer's characteristics are very similar.

Selection of suppliers: Evaluations must be made in order to select suppliers. Conventional or sharp classifications of suppliers, e.g. according to delivery date and quality characteristics, do not guarantee that a promising and long-term relationship can be maintained. The structure contained in the stored supplier data, but not visible or not yet visible, is made visible by fuzzy class formation. It allows a targeted and effective treatment of the individual supplier classes.

Business information systems generate a wealth of data. In day-to-day business, but above all to secure decisions, it is necessary to analyze these sometimes extensive data stocks or databases. Without a proper KDD process with fuzzy data analytics, you risk missing out on valuable information—both risks and opportunities.

Databases and Fuzzy Logic

In databases, especially in relational database systems, the characteristic values are assumed to be unique and queries to the database produce clear results. The feature values in the databases are precise, i.e. they are unique. Already in the requirement of the first normal form we require that the characteristic values are atomic and come from a well-defined range of values.

The characteristic values stored in a relational database are secure, i.e. the individual values are either known or unknown. An exception are the zero values, i.e. characteristic values that are not or not yet known. In addition, the database systems do not support us in any way in modeling an existing stochastic uncertainty. In other words, probability distributions for feature values are excluded; it remains difficult to express whether a given feature value corresponds to the true value or not.

Queries to the database are sharp. They always have a dichotomous character, i.e. a query value given in the query must either match or not match the characteristic values in the database. An evaluation of the database in which a query value "more or less" matches the stored characteristic values is not permitted.

With data modeling, a larger field of application can be opened up if incomplete, vague or imprecise facts are allowed. With the help of fuzzy logic, various model extensions were proposed, both for the entity relationship model and for the relational model. For example, in his dissertation, Chen expanded the classical normal forms of database theory into fuzzy ones by allowing fuzziness in the functional dependencies many different proposals for fuzzy data models for databases can be found in.

Investigations have also been made for the extension of relational query languages with fuzzy logic. For example, Takahashi proposes a fuzzy query language (FQL) based on the relational domain calculus. The language FQUERY by Kacprzyk and Zadrozny uses fuzzy terms and has been implemented as a prototype in the Microsoft product Access. In our work on a fuzzy classification and a fuzzy classification query language FCQL (fuzzy classification query language), we choose a slightly different research direction, originally indicated by Schindler: we limit ourselves to an extension of the relational database schema by creating a context model for propose fuzzy classification of table contents. Based on this, we develop the language FCQL, which allows fuzzy queries to the database schema using predefined linguistic variables and transmits them to the underlying (sharp) database in SQL calls.

In this way, we avoid migrating the database to a fuzzy database at great expense or confronting the user with fuzzy SQL. Fuzzy predicates would lead to a variety of semantic effects and a user would have to make different interpretations. With FCQL, on the other hand, we give the user a data mining tool so that he can start extended queries and calculate improved decision-making bases based on a predefined fuzzy classification of his data stocks.

Context Model and Classification

Extensive databases are often confusing and therefore difficult to analyze and evaluate. In order to obtain meaningful information, the user must restructure and, if necessary, condense their stocks. To this end, various methods and concepts for building and operating a data warehouse have been developed. There are also data mining tools to gain new insights from the databases. We choose a context model approach to be able to specify classes in the relational database schema. For the analysis and evaluation of a large number of suppliers, for example, it makes sense to group suppliers that are as similar as possible into classes. You then get the set of "suppliers with quality problems" or the set of "suppliers with whom the business relationship should be expanded" as an example. Such a combination of suppliers into classes means a reduction in complexity. The user can thus maintain and analyze his supplier relationships more clearly, thanks to the reduced flood of data.

In addition, important characteristics of the suppliers are made visible through the classification. This additional knowledge allows the user to analyze entire classes in a targeted and holistic manner and to work out the relationships between different classes.

When classifying objects in a relational database, a distinction can be made between sharp and fuzzy methods. In the case of a strict classification, the database objects are assigned to the class in a dichotomous manner, i.e. the set membership function of the object to the class is 0 for not included or 1 for included. A classic procedure would therefore assign a supplier to the "Suppliers with quality problems" class or the "Suppliers with whom the business relationship should be expanded" class. A fuzzy procedure, on the other hand, allows values between 0 and 1 for the quantity membership function: A supplier can, for example, belong to the class "suppliers with quality problems" with a value of 0.3 and at the same time to the class "suppliers with whom the business relationship has been developed" with a membership of 0.7 should be". A fuzzy classification therefore enables a differentiated interpretation of the class affiliation; With database objects of a class, one can distinguish between peripheral and core objects, and database objects can also belong to two different classes at the same time.

Context Redundant Tuples

In the relational model, one speaks of redundant tuples or tables if there are multiple occurrences in the tuple components that could be deleted without loss of information. The theory of normal forms was developed for the preservation of redundancy-free relations. The redundancy of two tuples in the context model is softer defined: Two tuples t and t' are called context-redundant if all tuple components t_i and t'_i belong to the same equivalence class. Context-redundancy-free relations are obtained by mixing operations, which are discussed in more detail below.

A database object belongs to a class if its feature vector points to the corresponding subdomain. In the presented context

model, the set-theoretic unification as a mixing operation is chosen as the classification function. This operation is performed when evaluating an expression of context-based relational algebra in order to obtain context-redundancy-free result relations.

The use of the minimum operator as an aggregation operator means that the characteristic with the lowest membership value is decisive for determining whether an object belongs to a class. When classifying suppliers, a supplier is acceptable in terms of its delay in delivery, but delivers poor quality. An aggregation with the minimum operator would only classify the supplier as a “poor” supplier because of its quality. If, on the other hand, the supplier evaluation takes place through human consideration, then one would make a certain compensation between poor quality and acceptable delay. In other words, in many cases

poor.quality will not be the only criterion when ranking a supplier

For code generation we had to extend the database schema with three descriptive tables for contexts, classes and membership functions. we have tested our implementation on an industry database. The responsible marketing department has a mature data Warehouse, but they want to further analyze and evaluate the customer and product inventories using methods of fuzzy logic. Thanks to the chosen approach with fuzzy classes, we didn't have to carry out a complex database migration to take over the data stock. With the field test, we hope that the classification language FCQL will complement the tool set for data mining and knowledge discovery in a meaningful way.