

Expert Database System for Converting Unfit Query to Standard SQL Query

Laith R. Flaih
Computer Science Department
Cihan University-Erbil
d.lath1974@yahoo.com

Abstract

Today, the integration of Expert System (ES) and Database Management System (DBMS) is an emerging research area that is expected to become one of the most important application development products. The proposed system characterized a new class of computer systems to be called Expert Database System (EDS) which is a combination of two systems. They are the Database system and Expert system.

This research considers that the database contains a lot of data, which is used by a large number of users with different computer education levels. It is a fact that the DBMS is always strict in query execution that the unfit query for a simple reason would be rejected, this rejection affects communication momentum and user disappointment by not having the desired data. The proposed system aims to prevent rejection of unfit query that is entered by the user through preparing it for optimization it so that it is in a simple correct form which is accepted by the DBMS.

Keywords: Expert System, DBMS, SQL, Database

Introduction

Database is required by efficient data management, is a shared, integrated computer structure that houses a collection of (Rob & Coronel, 2007):-

1. End user data: raw facts of interest to the end user.
2. Metadata: data about data, through which the data are integrated, such as data characteristics and relationship link.

Data-Raw Facts. Raw is used to indicate that the facts have not yet been processed to reveal their meaning. Such as a telephone number, a birth date, a customer name, a year-to-date (YTD) sales value, and so on. Data have little meaning unless they have been organized in some logical manner (Rob & Coronel, 2007).

An application program accesses the database by sending queries or requests for data to the DBMS. A query typically causes some data to be retrieved; a transaction

may cause some data to be read and some data to be written into the database, other important functions provided by the DBMS include protecting the database and maintaining it over a long period of time (Elmasri & Navathe, 2007).

Expert system (ES) is a decision-making or problem-solving software package that can reach a level of performance comparable to-or even exceeding-that of a human expert in some specialized and usually narrow problem area (Turban et al., 2005).

Shell is special-purpose tools designed for certain types of applications in which the user must supply only the knowledge base (Giarratano & Riley, 1998), the classic example of this is EMYCIN (Empty MYCIN) shell. This shell was made by removing the medical knowledge base of the MYCIN expert system. Clearly, the combination of the two technologies would benefit both expert and database systems. Expert system and AI technology in general, will contribute in database systems in areas such as providing a useful reasoning ability in query optimization tasks. DBMS technology will contribute to expert systems in giving them the ability to access the large collection of facts and also to apply features such as concurrency control, data security and optimized access to knowledge base items (Beynon-Davies, 1991).

In this paper we focus on query optimization by using Expert Database System (EDS). One major criticism of many early DBMSs has been their lack in of efficiency in handling the powerful operations they offer, particularly the content-based access to data by queries. Query Optimization tries to solve this problem by integrating a large number of techniques and strategies, ranging from logical transformations of queries to the optimization that optimization procedures either attempt to maximize output for a given number of resources or to minimize the resource usage for a given output. Query Optimization also tries to minimize the response time for a given query language, and minimizing cost times such as communication cost, secondary storage access cost, storage cost, computation cost (Jarke. & Koch, 1984).

Then the organization of this paper is as follows: the Expert System is described in section 2. SQL (Structured Query Language) explained in section 3. After this preprocessing and the overview of the propose system which describe the classification process explained in section 4. And the last section gives the conclusion of this paper.

Expert System

Expert system is a computer program that emulates the decision-making ability of a human expert (Rob & Coronel, 2007).

The term emulates means that the expert system is intended act in all respects like a human expert, the basic idea behind ES, an applied artificial intelligence technology, is simple. Expertise is transferred from the expert to a computer. This

knowledge is then stored in the computer, and users run computer from specific advice as needed (Turban et al., 2005).

The expert system asks for facts and can make inference and arrive at a specific conclusion, then, like a human consultant, it advises non expert and explains, if necessary, the logic behind the device (Turban et al., 2005).

The following properties possessed by a general expert system (Shannon, 1955):

1. The ability to explain and justify answers, on the basis of theory, or by citing relevant heuristic rules, or by appeal to past case histories.
 2. The closeness of reasoning procedures to those used by human experts.
 3. The ability to summarize and point out features of the problem situation that were most important in leading to an answer, including information about which other factors might still have an effect, if they were to become known.
 4. The ability to grow gradually by adding new pieces of knowledge, usually in the context of solving unfamiliar problem. (Leondes, 2002)
- Three major components that appear in virtually every expert system are the knowledge base, inference engine, and user interface we see in figure 1 (Beynon-Davies, 1991).

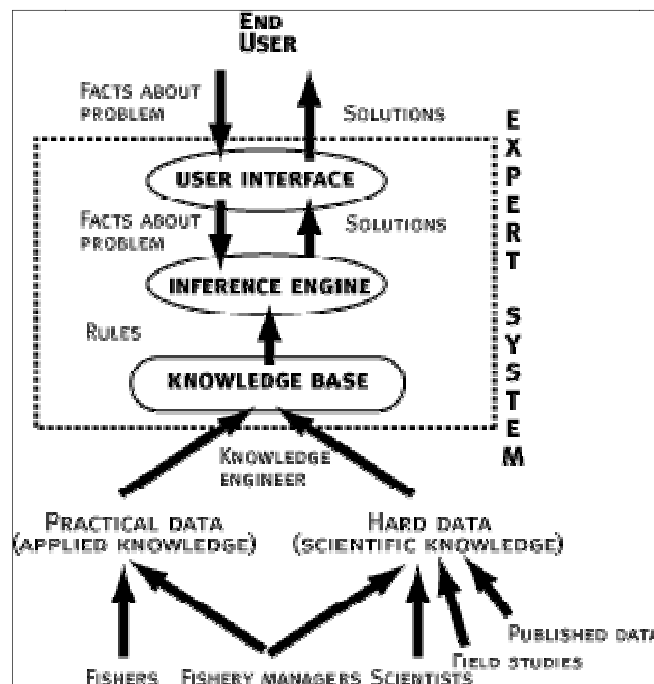


Fig. 1. Expert System

Knowledge Base

The knowledge base contains the relevant knowledge necessary for understanding, for manipulating, and solving problems (Turban et al., 2005).

Knowledge base includes two types of knowledge. One is analytical knowledge that consists of analytical process models (e.g., differential equations), estimation methods for parameters and state variables as a basis for feature extraction, and normal process behavior. The second is heuristic knowledge that is constructed by controllers and their parameter trees (connection of process fault symptoms and tailed controllers), historical records (e.g., biomass, substrate, and product concentrations in different culture times in a fermentation process), and process state statistics (e.g., fault and malfunction probabilities). The historical data of a process contain important information about the operation of the process, but they are not in a useful form for understanding the process, and traditional data analysis tools cannot make use of these huge data sets (Leondes, 2002).

Inference Engine

An inference engine is a procedure that uses items in the knowledge base of an expert system to draw conclusions and solve problems (Leondes, 2002).

The “brain” of the ES is the inference engine, also known as the control structure or the rule interpreter (in rule-based ES). This component is essentially a computer program that provides a methodology for reasoning about information in the knowledge based on the blackboard, and for formulating conclusions. This component provides directions about how to use the system’s knowledge by developing the agenda that organizes and controls the steps taken to solve problems whenever consultant takes place. In the production system, the inference engine performs the recognize-act control cycle. The producers that implement the control cycle are separate from the production rules themselves. Two popular approaches for inferencing are forward chaining and backward chaining (Luger, 2009; Turban et al., 2005).

User Interface

User interface is the mechanism by which the user and expert system communicate.

Expert systems contain a language processor for friendly, problem-oriented communication between the user and computer. This communication can best be carried out in natural language. Due to technological constraints, most existing systems use the question-and-answer approach to interact with user. Some times it is supplemented by menu, electronic forms, and graphics (Giarratano & Riley, 1998).

Shell

Shell is special-purpose tools designed for certain types of applications in which the user must supply only the knowledge base, the classic example of this is EMYCIN (Empty MYCIN) shell. This shell was made by removing the medical knowledge base of the MYCIN expert system (Giarratano & Riley, 1998).

A shell is an expert system without the domain-specific knowledge. Here we can make the analogy with database management system. A DBMS is a tool for building database system, A shell is a tool for building expert systems, Just as a DBMS is built upon some underlying data model, so an expert system shell is build some underlying formalism knowledge representation(Giarratano & Riley, 1998; Beynon-Davies, 1991).

SQL (Structured Query Language)

SQL is referred to as a non-procedural database language. SQL does not make a DBMS

It is just a medium which is used to as a means of communicating to the DBMS what you want it to do (Din, 1994). What this means is that when you issue an SQL command to retrieve data from a database, you do not have to explicitly, tell SQL where to look for the data. It is enough just to tell SQL what data you want to be retrieved. The DBMS will take care of locating the information in the database. This is very useful because it means that users do not need to have any knowledge of where the data is and how to get at it, Procedural languages such as COBOL or Pascal and even older databases based on the network and hierarchical data models require that users specify what data to retrieve and also how to get at it. Most large corporate databases are held on several different computers in different parts of the building or even at different geographic locations. In such situations, the non-procedural nature of SQL makes flexible, ad hoc querying and data retrieval possible. Users can construct and execute an SQL query, look at the data retrieved, and change the query if needed all in a spontaneous manner. To perform similar queries using a procedural language such as COBOL would mean that you would have to create, compile and run one computer programs for each query (Din, 1994), SQL has two parts (Zainvi, 2017):-

1. Data Definition Language (DDL): to create, alter, or drop tables and indexes.
2. Data Manipulation Language (DML): to insert, update, retrieve or delete the data in the tables.

The Proposed System

The proposed system is a new class of computer systems to be called Expert Database System (EDS), which a combination of two systems they are the database

system and expert system. The database contains a lot of data, which is used by a large number of users with different computer education level. Our system aims to prevent the rejection of unfit query that is entered by the user through preparing it for optimization, So that it is correct from which accepted by DBMS, as we shown in figure 2:

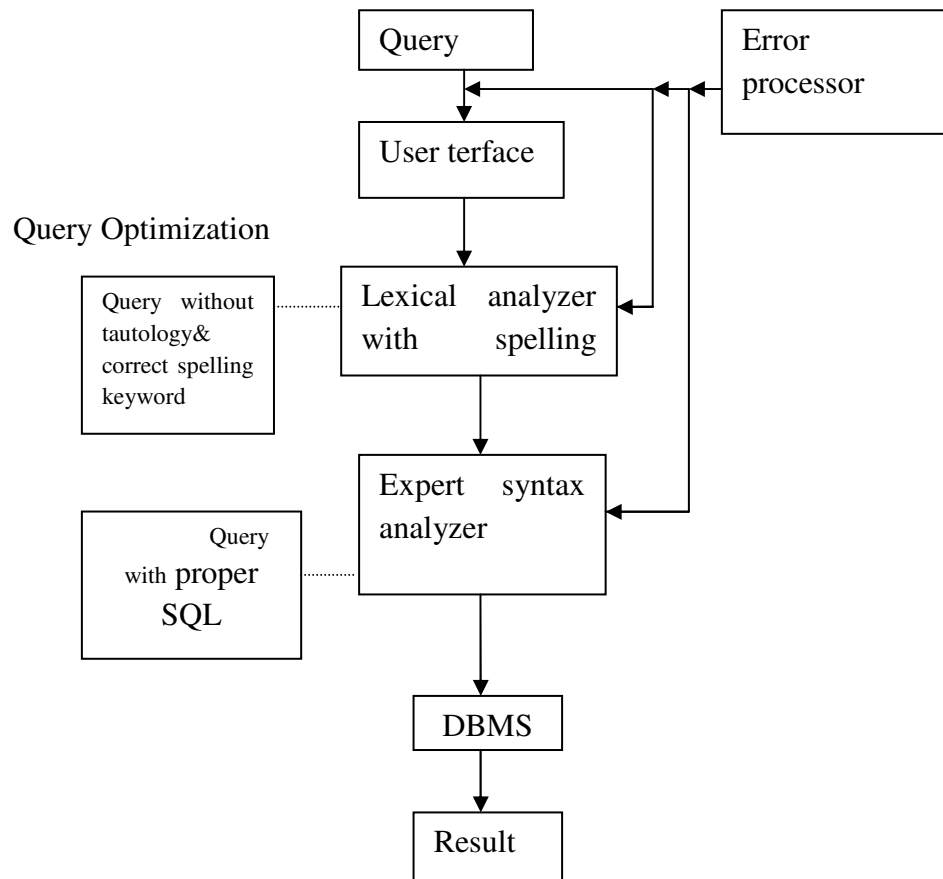


Fig. 2. Expert Database System for Query Optimization

The proposed system consist a dictionary file in which the dictionary file consists a large number of SQL command used in database “such as select, delete, drop ...etc”, it is worth mentioning all command words are arranged alphabetically.

Our system consists of two phases, first phase users understand SQL, second phase user doesn’t understand SQL using (natural language).

Users Understand SQL

in which user in the application know the basic knowledge of the SQL, this type of users understand SQL to prevent rejection of unfit query that is entered by the user and simply correct it, then accepted by the DBMS and gives the result.

Users understand SQL consists of two types of analysis, first is lexical analysis and second is syntax analysis.

Lexical Analysis with Spelling Corrector

Lexical analysis is a process where the stream of characters making up the source program is read from left-to-right and grouped into tokens, tokens are categorized to the rules as a symbol (identifier, number, comma,...etc), the process of forming tokens from an input stream of character is called tokenizing, after tokenizing SQL query statement, our proposed system can take the SQL command word and check whether this SQL command is correct or not. In this analysis we will take SQL command and compare SQL command with dictionary file command.

The proposed system consists a dictionary file in which the dictionary file consists a large number of SQL commands used in database system "such as select, delete, drop...etc", it is worth mentioning all command words are arranged alphabetically.

After comparing SQL command with SQL dictionary command, if this command word is correct then it will immediately be accepted by the DBMS, but if the mentioned SQL command is wrongly spelt then it will be rejected by the DBMS. In our system, if any SQL command is taken, it will be compared with correct SQL command that have been established in the dictionary, in case the command is misspelled and not found in the dictionary file, then the process will correct the command and make it meaningful spelled command word to dictionary, and misspelled command will be added to dictionary file, and the all founded misspelled command is corrected Automatically and the system will reject it from adding it to the dictionary, as we show in Figure 3.

Expert Syntax Analyzer

After checking the command words must be checked syntactically also, syntax analysis is a process of analyzing a text, made of a sequence of token to determine its grammatical structure. Our syntax analyses are:

1- Diagnosis of the incorrect keyword

A- If the user ask for a field which is not used in database.

Our system will check field name, table name and database name for knowing whether this field exists or not. In case the field was not found the proposed system gives a message to user stating this field is not found in database.

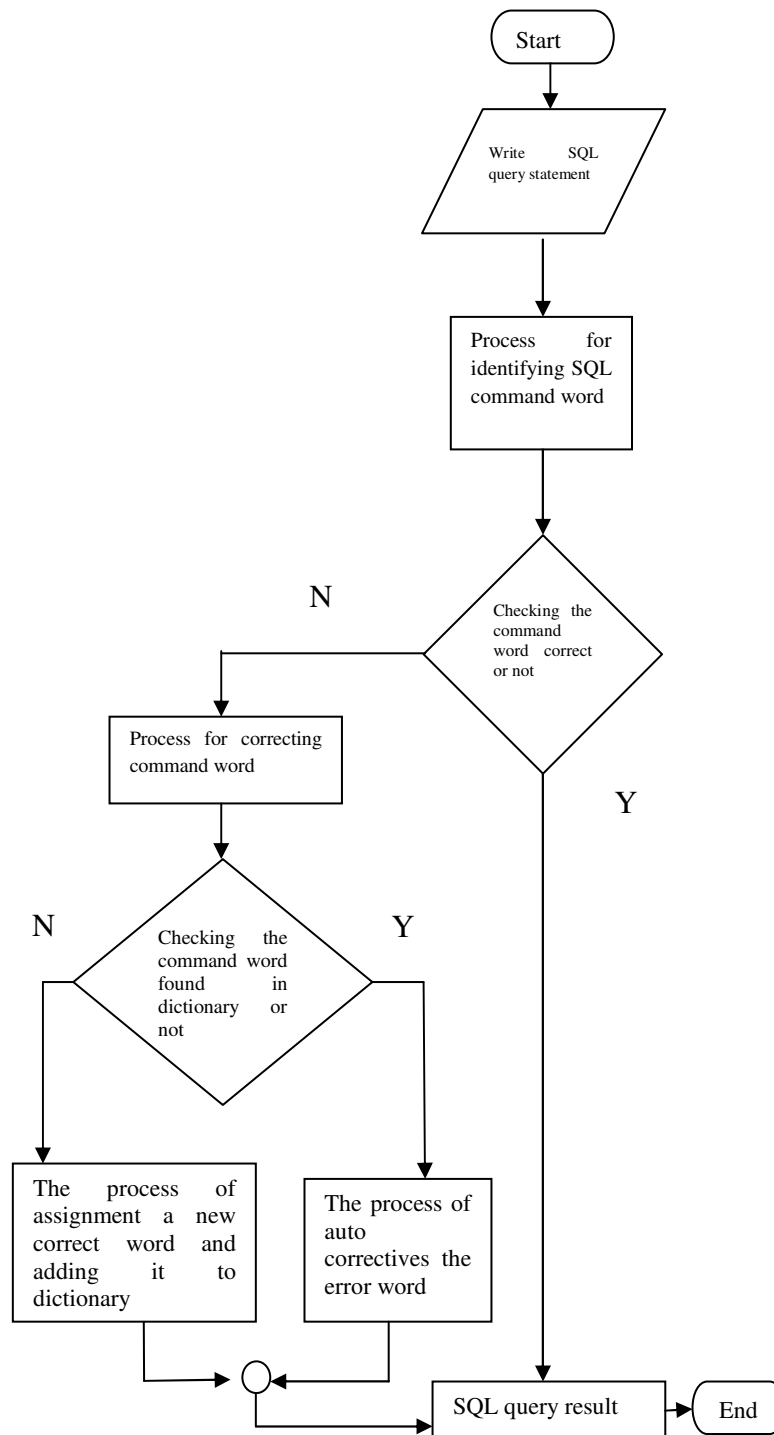


Fig. 3: Flowchart for Lexical analysis with spelling corrector

B- If the user ask for a command which is not used.

Our system will check SQL command if it is used in database or not, the proposed system will give a message to user informing, this command word is not used in database.

C- Insert field(s) value(s) without inserting record.

Inserting values into table, at the same time this value is not compatible with field data type, our system give a message to user saying, type mismatch.

2- Making a decision about the related field to the right relation.

Asking user for a field in a table, but asked field is not found in that table and found in another table, our system gives a message to user by informing, this field not found in that table but found in another table after that correcting query and give a result to user.

3- Completing the incomplete query by adding the necessary missing keyword.

Incomplete query entered by the user completed by adding missing keywords, our proposed system gives a message to user say "the entered query is incomplete", after completion it will give result to user from the data base.

4- The resulted query from previous process would be converted to the standard SQL.

Converting all previous queries to standard SQL query statement, in order to be understandable and acceptable by the DBMS

Conclusions

- 1- Dealing with "database" is done through persons at different education levels, thus dealing with those variables causes errors that will cost time and efforts. The proposed system reserve time and efforts through capabilities to implement erroneous "query".
- 2- The proposed system has potential to deal with "natural language" that means the rate of mistakes will definitely be low and dealing with the "Database" does n't require high skills and efficiency that enables the user to make use of the natural language.
- 3- The system enables saving treating time through its capability in dealing with erroneous "query".

References

- Beynon-Davies, P. (1991) *Expert Database Systems A Gentle Introduction*. UK: McGRAW-HILL Book Company.
- Din, A.I. (1994) *Structured Query Language (SQL) A Practical Introduction*. NCC Blackwell.
- Elmasri, R., Navathe, S. B. (2007) *Fundamentals of Database System*. Fifth Edition, UK: (Addison Wesley) Pearson Education
- Giarratano, J. & Riley, G. (1998) *Expert Systems Principles and Programmings*. Third Edition, Boston: PWS Publishing Company.
- Jarke, M. & Koch, J. (1984) Query Optimization in Database System, *Computing Surveys*, 16(2).
- Leondes, C.T. (2002) *Expert Systems*, Academic Press.
- Luger, G.F. (2009) *Artificial Intelligence Structures and Strategies For complex Problem Solving*. Sixth Edition, UK: (Addison Wesley) Pearson Education.
- Rob, P. & Coronel, C. (2007) *Database Systems: Design, Implementation, and Management*. Seventh Edition, USA: Thomson Course Technology.
- Shannon, C. E. (1955) A chess playing machine. In *The World of Mathematics*, Vol. 4 (J. R. Newman, Ed.). New York: Simon and Schuster.
- Turban, E., Aronson, J. E. & Liang, T.P. (2005) *Decision Support System and Intelligent Systems*, Seventh Edition, UK: (Prentice Hall) Pearson Education
- Zainvi, S. F. (2017) How To Use Structured Query Language (SQL). <http://www.howtodothings.com/computers-internet/how-to-use-structured-query-language-sql>. [Accessed on July 18th 2017]