



Project Case Study

Provider Name: KALKI Communication Technologies

Client Name: Qsupport

Project Title: *Expert System Development*

PART I: The Problem

Qsupport is a customer support company. Personal as well as corporate customers send queries on various software and hardware related problems by e-mail. Qsupport has got a support team, members of which will go through the e-mails and reply the questions posed by the customers. In case the support person does not have an answer at hand, he will go through appropriate reference material and find out the answer. Since all questions asked and answers sent were stored in a database, this database was the primary reference material. Qsupport was using a simple word search program to search this database and find appropriate questions. The database grew in size and contained around 15,000 questions and their answers. This sharply brought down the accuracy of the keyword search system and the cycle time between receiving a query and answering it (which is a crucial performance parameter for Qsupport) started to go up. It was to counter this that Qsupport turned to Kalki for an expert system.

PART II: The Solution

The objective was to cut down the time it took to locate questions and answers in the database once a query was received. The problem with a simple keyword search was that too many matches would be found requiring the user a good amount of time to go through each of them. So to develop a more sophisticated search was the primary objective. At the same time some automation of the answer retrieval was desired so that cycle time in answering a query could be further reduced. The end result was an expert system which when prompted with a new query, will search the existing database and find out the most accurate questions (and their answers). Since the whole text of the e-mail was fed to the expert system, it was not required of the user to identify and type in keywords to be searched. Since the expert system used an advanced algorithm for searching incorporating Natural Language Processing methods, the results were accurate. The results were confined to the best four matches (which was found sufficient due to the high accuracy) and this minimized the time required of user to go through them. The expert system provided an interface for the user to combine suitable responses from those retrieved and send it back so that time required in formulating the answer was reduced. Since the whole expert system was developed as an ActiveX dll, it could be easily incorporated into any other system (including a web page) without much change. The expert system also provided various statistical information regarding its own performance as well as on the queries asked by various users and their domains/sub-domains. This helped the client in planning for resources required to provide the relevant expertise.

Description:

First part of the solution consisted in developing a systematic storage of the questions and their answers rather than storing all of them together in simple text form. The expert system uses a well defined hierarchy of domains and sub-domains for storing the questions. Also each question was provided an indexing mechanism based on keywords present in it that are most relevant to the domain and sub-domain of the question as well as the question itself. Each index was assigned a correlation factor with respect to the domain, sub-domain and the question. The expert system was programmed so that it will automatically modify these correlation factors (based on its performance). When a new query was posed to the system, it will convert the question (which consists of the entire e-mail sent by a user) to a format suitable for searching against the database indices. This required the system to identify phrases in the question (such as Visual Basic which will be treated as a single word rather than two separate words), convert acronyms present in the mail (like VB to Visual Basic), use morphology rules to find the base forms of various words (like installing and installation to install) and remove all the junk words that are not relevant to the query. After this, the expert system uses the index information in the database to identify the most likely domain and sub-domain for the query. Also it finds the best four matching questions with their answers for the query and presents them to the user. The expert system has got a user feedback mechanism with which he can evaluate the responses provided by the system as an exact match, satisfactory response or a mismatch. In case of a mismatch the system allows the user to go into a normal keyword search or a browse mode to search the database. For satisfactory response, the system allowed the user to combine the answers provided and transfer it to another application (from where it can be sent back to the customer). For cases where the system fails to retrieve an answer, the query will automatically get stored by the system so that the user (or somebody else) any answer it at an off-peak time. An administration module was also provided for the expert system. An administrator could validate answers provided by other users for questions (which the expert system failed to answer) before they get moved to the database. He could also get a snap shot of the entire question-answer database with its various domains and sub-domains so that he can assess knowledge base in his organization exactly. He could also see daily, weekly or monthly statistics regarding the expert system performance as well as the queries raised by the users. Later a security mechanism was added to the expert system so that the administrator can restrict access to the expert system for various user based on different information accounts. This was done keeping in view a single installation of the expert system which can be used by different organizations without fear of compromising information regarding their products and features to others. Since bulk of the logic executed by the expert system is implemented as stored procedures, the performance is high and the system is scalable. And since the system is developed as an ActiveX dll, it can be easily integrated to other applications including web pages.

Technology Used

Front end was developed in Visual Basic 6.0

System uses an MS SQL Server 7.0 backend (and hence T-SQL is used for backend procedures and jobs)

Expert system uses a custom dictionary and thesaurus for Natural Language capabilities

Standard morphology rules provided in English are used to obtain root forms of words

All NLP modules were developed as separate COM objects for easy reusability

Dates:

Start: 26/06/2000

End: 14/08/2000

Duration:

6 weeks * 2 person = 12 manweeks (3 manmonths) for the initial version
2 weeks * 1 person = 2 manweeks (.5 manmonth) for the security add-on
2 weeks * 2 person = 4 manweeks (for testing and review)

Location of work: *(The city, country and company in which the work was executed)*

KALKI Communication Technologies
Bangalore
India

Team Size: *(The number of people who worked on the project. Fill out the table below.)*

SR No.	Role Title	Name	Yrs. Experience	% Involvement
1	Project Lead	Sreeni T S	3	50
2	Developer	Jeyapandyan	1	30
3	Testing & QA	Prashant SM	3	20

PART III:

PART IV: About the Client

Qsupport is 24/7 online support company which provides online solutions for any computing and internet problem. Located in Bangalore, India the company operates in two modes. For personal users accessing services for their home PCs the support provided is free. Revenue comes through corporate customers for whom qsupport provides online support. The company has got a good support team that works round the clock to answer queries posted by customers and has built a huge database of questions and answers pertaining to various domains. More information can be had at <http://www.qsupport.com>