

SELECT attributes FROM sourceTableName WHERE (condition1,...,condition.n) is Q1 [(condition1...,conditionn) is Q2 WEIGHT weight1, weight2,...,weightn WITH $\alpha_1, \alpha_2, \dots, \alpha_n$ FOR β .

WEIGHT: importance defined by user; WITH: thresholds of different predicates, FOR: defined here as threshold of fuzzy linguistic quantifies; $\alpha_1, \alpha_2, \alpha_n, \beta$ can be precise or fuzzy items. When Q1, Q2=all or any, the query will be translated to traditional query connected with AND or OR. There are many different methods to define membership functions (e.g. young, height and etc.), Here, we take membership functions of predicates as:

$$\mu_{\text{young}}(\alpha) = \begin{cases} 1 & \alpha \leq 20 \\ 1 + \left(\frac{\alpha-20}{7}\right) & \alpha \geq 20 \end{cases}$$

$$\mu_{\text{tall}}(\alpha) = \begin{cases} 1 & \alpha \geq 100 \\ 1 + \left(\frac{\alpha-100}{10}\right) & \alpha < 100 \end{cases}$$

$$\mu_{\text{salary-height}}(\alpha) = \begin{cases} 1 & \alpha \geq 7000 \\ \left(\frac{\alpha}{7000}\right) & 0 \leq \alpha \leq 70000 \end{cases}$$

$$\mu_{\text{most}}(\alpha) = \begin{cases} \frac{\alpha}{0.5} & 0 \leq \alpha \leq 0.5 \\ 1 & \alpha > 0.5 \end{cases}$$

Query searching Stu-Id, Name, Age, Height and F-Name of Student in the relation “Student” that meets most of conditions “young, better, Height” are true. Weights are given as: Age (0.8), Height (0.5), Best (0.6); Thresholds: Age (0.7), Height (0.4), Best (0.5); matching degree (0.6). Then we make SQL query as:

SELECT Stu-Id, Name, Age, Height, Grad

FROM Student

WHERE (Age is young, Height is tall, Best performance) is most

WEIGHT 0.8, 0.5, 0.6

WITH 0.7, 0.4, 0.5

FOR 0.6

Through usage of positive method, it will get membership degree of attributes, and then gain the results according to matching degree through combination of SQL functions and Zadeh's Computational Approach. All the relative Tables in the process of computation are as follows: Table 1.1: basic relation; Table 1.2: initial membership degree of attributes; Table 1.3: membership degree of attributes combined with weights.

Stu-Id	Name	Age	Height (cm)	Salary
CG-2001	Samyak Gupta	3	172	8000
CG-2002	IshaSahu	4	183	3500
CG-2005	ThanishaYadav	5	183	3500
CG-2006	GulshanSahu	20	175	6000
CG-2007	Ruanshyadav	22	185	4900
CG-2008	Shahil Sharma	11	165	4200
CG-2009	Bhveshsingh	20	173	4800

Table 1.1: Student Table

Through the help of matching degree of young, tall, salary-high with weights and SQL functions, (when $\mu_{young} \in [0.7, 1, NULL]$); (when $\mu_{tall} \in [0.4, 1, NULL]$);

Stu-Id	μ_{young}	μ_{tall}	$\mu_{salary-height}$
CG-2001	0.961	0.236	1.0
CG-2002	0.281	0.552	0.5833
CG-2005	1	0.671	0.5833
CG-2006	0.1	0.308	1.0
CG-2007	1	0.8	0.8167
CG-2008	0.410	0.138	0.8667

CG-2009	1	0.257	0.8
---------	---	-------	-----

Table 1.2:Initial membership degree of Age, Height, Salary of each tuple

Stu-Id	μ' young	μ' tell	μ' salary-height	μ
CG-2001	0.8(1)	0.236(NULL)	1.0	0.667
CG-2002	0.281(NULL)	0.5(1)	0.5833	0.667
CG-2005	0.8(1)	0.5(1)	0.5833	1.000
CG-2006	0.1(NULL)	0.308(NULL)	1.0	0.333
CG-2007	0.8(1)	0.5(1)	0.8167	1.000
CG-2008	0.410(NULL)	0.138(NULL)	0.8667	0.333
CG-2009	0.8(1)	0.257(NULL)	0.8	0.667

Table 1.3:Membership degree of Age, Height, Salary combined with weights

μ_{most} is the matching degree of every tuple that satisfying the query. Here, we just give a simple example to represent the basic principle of fuzzy SQL queries with weights based on fuzzy logic conjunctions. Obviously, this strategy can be extended to more fuzzy queries and also can be extended to web retrieval.

Stu-Id	Name	Age	Height (cm)	Salary	μ_{most}
CG-2005	ThanishaYadav	5	183	3500	1.000
CG-2007	Ruanshyadav	22	185	4900	1.000
CG-2001	Samyak Gupta	3	172	8000	0.667
CG-2002	IshaSahu	4	183	3500	0.667

CG-2009	Bhveshsingh	20	173	4800	0.667
---------	-------------	----	-----	------	-------

Table 1.1: The Result Sets

In this paper, The working of Fuzzy Semantic Natural Language Processing algorithm and Fuzzy Fuzzy Query with Weights Based on Fuzzy Logic Conjunctions Algorithm is explained. Different type Parsing process and all possible steps for the Fuzzy Based Query System are discussed. In Fuzzy Natural Language Processing first built some set of rules and with the help of Fuzzy set we can compare the member function.

Conclusion:

Fuzzy Based Query Processing Language is a powerful tool for virtually interface provide in SQL database system. This Fuzzy Based Query Processing Language is currently used to handling simple Queries with standard join conditions. This all also support the SQL Queries for optimization. The optimization of Query from one single and simple statement is desirable to fetch more and more knowledge to understanding the language in easy way. The proposed system is very helpful to recognize the Query in term of Simple Natural Language related to common human being communication. This system is useful for those people and organization who working in the database system for query optimization and finally it's give the result in the form of database Queries and specially those users, who doesn't have good skill knowledge of database system. This work helps users to access data from database in easily manner. This proposed system is very efficient research to encourage the research work in Fuzzy Based Query Processing and to support each type of communication in Natural Languages.

The conventional Query in relational database management system is not capable of satisfying the needs for dealing with Queries which are in Natural Language. In this work, we have deliberated about Fuzzy based Query System and we have proposed an intelligent Fuzzy based Query optimization system based on Natural Language Processing which uses Fuzzy logic for Query Processing for Complex Queries. Now we have implemented a software tool which can execute Complex Query and easily fetch information from database.

References:

1. Miroslav, Hudec. (2009), An Approach to Fuzzy Database Querying, Analysis and Realisation. ComSIS Vol. 6, No. 2. . pp. 1-5.
2. G.R., Bamnote. Abhijeet, R. Raipurkar. (2013), Fuzzy Logic Based Query Optimization in Distributed Database. International Journal of Innovative Research in Computer and Communication Engineering. Vol. 1, Issue 2 : 2320 – 980
3. Jasmeen, Kaur., Bhawna, chauhan., Jatinder, Kaur. (2013), Implementation of Query Processor Using Automata and Natural Language Processing. International Journal of Scientific and Research Publications, Volume 3, Issue 5.
4. Poria, Soujanya., Basant, Agarwal. (2013), Dependency-Based Semantic Parsing for Concept-Level Text Analysis. 15th International Conference on Intelligent Text Processing and Computational Linguistics, Part I. Lecture Notes in Artificial Intelligence. pp. 113–127.
5. Dilek, Hakkani., Ali, El-Kahky., Liu ,Xiaohu. Sarikaya. (2014), Extending Domain Coverage Of Language Understanding Systems Via Intent Transfer Between Domains Using Knowledge Graphs And Search Query Click Logs. IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP).
6. Javubar, K., Sathick, Jaya A. (2014), Natural Language to SQL Generation for Semantic Knowledge Extraction in Social Web Sources. Middle-East Journal of Scientific Research 22 (3): 375-384 .
7. Soujanya, Poria. And Chen, Gui. (2014), A Rule-Based Approach to Aspect Extraction from Product Reviews. Proceedings of the Second Workshop on Natural Language Processing for Social Media (SocialNLP), PP 28–37.
8. White, Bebo. And Erik, Cambria. (2014), Jumping NLP Curves: A Review of Natural Language Processing Research. IEEE Computational Intelligence Magazine. pp. 410–415.
9. Maio, De Carmen., Fenza ,Giuseppe., Loia, Vincenzo. (2015), Natural Language Query Processing Framework for Biomedical Literature. 16th World Congress of the International Fuzzy Systems Association (IFSA). pp. 108-123.
10. Ramakrishnan, R. And Gehrke, J. (2000), Database Management Systems Third Edition. McGraw Hill, 65-84.

11. In-Su, Kang., Jae-Hak, J. And Jong-Hyeok, Lee. (2002), Database Semantics Representation for Natural Language Access. Department of Com Computer Science and Engineering, Electrical and Computer Engineering Division Pohang University of Science and Technology (POSTECH) and Advanced Information Technology Research Center (AITrc): 122-156.