

DESAIN FUZZY QUERY PADA BASIS DATA RELASIONAL

Ida Ayu Putu Febri Imawati¹⁾

I Nyoman Sumerta Yasa²⁾

^{1,2)} Program Studi Teknik Informatika
Denpasar, Indonesia
Universitas PGRI Mahadewa Indonesia

e-mail : febri@mahadewa.ac.id

Abstract

An application that uses a relational database generally defines, stores and performs retrieval of definite and clear data and information. Because of complex data and information needed by the user, sometimes user will have or even need incomplete, unclear and ambiguous data or information. This cannot be accommodated by SQL which is the standard relational database query language. The fuzzy technique integrated with the database is the answer to the user's needs. Therefore, in this study a fuzzy query is designed that transforms human natural language or linguistic variables into a SQL query language from a relational database. Using membership function of the fuzzy technique, the user will get a range of answers from the ambiguous data or information. The results of this research can be implemented by developing a special front end for users to perform queries. Thus, the existing relational database can still be used and its utilization can be increased.

Keywords: *fuzzy query, variabel linguistik, membership function, fuzzy relational database*

Abstrak

Sebuah aplikasi yang menggunakan basis data relasional umumnya mendefinisikan, menyimpan dan melakukan retrieval terhadap data dan informasi yang pasti dan jelas. Dengan semakin kompleksnya data dan informasi yang dibutuhkan oleh user, terkadang user akan memiliki bahkan membutuhkan data ataupun informasi yang tidak lengkap, tidak jelas dan ambigu. Hal ini belum dapat diakomodasi oleh SQL yang menjadi bahasa standar query basis data relasional. Teknik fuzzy yang diintegrasikan dengan basis data merupakan jawaban kebutuhan user tersebut. Karena itu dalam penelitian ini didesain fuzzy query yang mentransformasikan bahasa alamiah manusia atau variabel linguistik ke dalam bahasa SQL query dari basis data relasional. Dengan adanya membership function dari teknik fuzzy, user akan mendapatkan range jawaban dari data ataupun informasi yang ambigu tersebut.

Hasil dari penelitian ini dapat diimplementasikan dengan membuat front end khusus bagi user untuk melakukan query. Dengan demikian basis data relasional yang sudah ada saat ini masih tetap bisa digunakan dan pemanfaatannya bisa ditingkatkan.

Kata Kunci: fuzzy query, variabel linguistik, membership function, fuzzy relational database

1. PENDAHULUAN

Pada umumnya basis data standar dikelompokkan berdasarkan seperti apa data tersebut dipandang oleh user. Kemudian query digunakan untuk mencari data serta memanipulasi data yang dibutuhkan. Namun pada aplikasi nyata (*real world*) sering data ataupun informasi yang dimiliki kurang lengkap, mengandung ketidakpastian bahkan cenderung ambigu [1]-[7]. Hal ini yang menyebabkan penggunaan basis data standar tidak dapat mengakomodasi kebutuhan pemodelan data atau informasi kompleks yang tidak presisi dan ambigu. Basis data tradisional tidak mensupport ketidakpastian informasi.

Tool-tool tradisional yang digunakan pada saat komputasi, *crisp*, deterministik dan pasti. Kepastian akan menunjukkan bahwa struktur dan parameter model sudah diketahui dengan pasti. Pada situasi nyata, model tidak *crisp* dan tidak deterministik. Karena itu tidak dapat dijelaskan dan digambarkan dengan tepat. Dengan pengintegrasian teknik fuzzy di basis data maka akan dapat mendekatkan sistem dengan kehidupan manusia. Sistem lebih *user friendly*. Sistem akan dapat mengakomodasi bahasa sehari-hari manusia seperti banyak, sedikit, hampir, tua, muda, sedang, rata-rata dan yang lainnya. Ada dua cara untuk mengimplementasikan logika fuzzy ke dalam basis data yaitu dengan fuzzy query pada basis data dan dengan *fuzzy extention database* [7]. Cara *fuzzy extention database* yaitu dengan memasukkan informasi fuzzy ke dalam basis data. Namun basis data fuzzy ini masih belum begitu populer karena kebanyakan orang masih enggan mengganti data *crisp* mereka dengan data fuzzy sebelum mereka yakin [1]. Teknik pada saat melakukan query menggunakan fuzzy preferences dan/atau menggunakan kondisi fuzzy [4] dikembangkan untuk

me-retrieve informasi yang tidak pasti. Teknik yang diterapkan pada penelitian ini berjalan sebagai *middleware* [1]. Tujuan utama teknik ini adalah mengeksplorasi fasilitas atau fitur standar yang ada pada DBMS moderen. Relasional basis data yang konvensional masih digunakan, namun dibuatkan sebuah *front end* yang akan mengijinkan query dengan teknik fuzzy terhadap basis data. Dalam hal ini basis data tetap dalam keadaan *crisp*, teknik fuzzy hanya diterapkan di *front end* [7].

Program Studi Teknik Informatika di Fakultas Teknik dan Informatika Universitas PGRI Mahadewa Indonesia selain menggunakan sistem informasi akademik berbasis web juga memiliki sebuah aplikasi sistem akademik dan keuangan dengan nama SIAKKE yang berbasis dekstop (offline). Aplikasi ini dibuat menggunakan Microsoft Access 2003 dan basis data-nya menggunakan MYSQL. Aplikasi ini tetap dioperasikan dengan maksud sebagai *backup*. Bagian akademik sering membutuhkan retrieval informasi dari basis data ini. Tabel mahasiswa merupakan salah satu tabel yang sering di-*retrieve* informasinya. Mahasiswa prodi teknik informatika terdiri atas mahasiswa regular dan transfer. Mahasiswa transfer berasal dari mahasiswa diploma yang akan melanjutkan ke jenjang strata 1 ataupun dari jenjang strata 1 yang pindah dari perguruan tinggi sebelumnya. Untuk mengambil mata kuliah di prodi teknik informatika mahasiswa transfer ini memiliki nilai yang dibawa dari perguruan tinggi sebelumnya kemudian dikonversi. Hasil konversi matakuliah ini berupa SKS yang diakui beserta nilainya. Masing-masing perguruan tinggi memiliki kurikulum yang berbeda-beda karena itu hasil konversi berbeda untuk perguruan tinggi yang satu dengan yang lainnya.

Mahasiswa sering membahasakan jumlah SKS ini seperti: jumlah SKS yang diakui ‘banyak’, ‘sedikit’, ‘rata-rata’ atau ‘sedang’. Pada aplikasi SIAKKE informasi tersebut masih berupa *crisp*. Karena itu, penelitian ini akan mendesain fuzzy query untuk melakukan retrieval informasi tersebut sehingga bagian akademik dan mahasiswa akan mendapatkan informasi sesuai kebutuhan serta lebih mendekati ke keadaan yang lebih nyata. Hasil penelitian ini dapat diimplementasikan dengan menambahkan *interface* fuzzy query pada aplikasi yang sudah ada. Lebih jauh hasil penelitian ini akan dapat meningkatkan pemanfaatan basis data relasional yang sudah digunakan hingga saat ini.

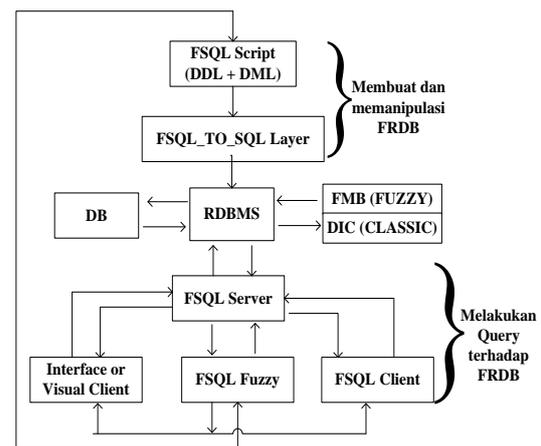
1.1 Variabel Linguistik

Umumnya data yang disimpan di basis data adalah *crisp* namun permintaan terhadap data atau informasi dalam situasi yang nyata kemungkinan fuzzy. Untuk mengakomodasi permintaan tersebut diperlukan sebuah query untuk mengakses basis data, *Relational Database Management System* (RDBMS) mengalami kesulitan untuk menangani hal tersebut. Misalnya sebuah query: “Mahasiswa transfer yang merupakan mahasiswa baru dengan SKS yang diakui sedikit”. SQL tidak dapat secara langsung memproses query tersebut karena mengandung ketidakpastian, dan ambigu seperti mahasiswa baru dan SKS yang diakui sedikit. Bahasa alamiah seperti mahasiswa baru, SKS yang diakui sedikit mengandung term dasar yang disebut sebagai *atomic term*. Contoh *atomic term* adalah lama, baru, banyak, sedikit, sedang. Kumpulan dari *atomic term* disebut sebagai *composite term* contohnya SKS yang diakui paling sedikit. Pada teknik himpunan fuzzy *atomic term* disebut sebagai variabel linguistik [1]. Variabel linguistik berbeda dengan variabel numerik, isi dari variabel linguistik berupa kata atau kalimat.

Tujuan penggunaan variabel linguistik ini adalah memberikan pendekatan terhadap suatu fenomena yang tidak terdefinisi dengan baik. Pada linguistik variabel ini biasanya dimodifikasi dengan menambahkan kata sifat seperti lebih, kurang, paling, hampir, sangat. Kata-kata tersebut disebut sebagai *linguistic hedges* [1].

1.2 Arsitektur Fuzzy Relational database Management Systems (FRDBMS)

FRDBMS adalah sebuah basis data yang menyediakan banyak fungsi seperti deskripsi data (DDL atau *Data Definition Language*), manipulasi data (DML atau *Data Manipulation Language*), perawatan integritas dari *Fuzzy Relational Database* (FRDB) menggunakan aturan-aturan integritas, kerahasiaan dengan menerapkan hak akses, manajemen akses, keamanan dan bantuan terhadap user [8]. Arsitektur FRDBMS ini seperti pada gambar 1.



Gambar 1. Arsitektur FRDBMS

Arsitektur baru dari FRDBMS ini berdasarkan GEFRED (*GEneralized model Fuzzy heart Relational Database*) model [9]. FSQL digunakan sebagai DDL dan DML. FRDBMS dapat memodelkan *Relational Database* (RDB) dan juga *Fuzzy Relational Database* (FRDB). Layer Software akan mentransformasikan *command* yang dituliskan user di bahasa

FSQL yang ekuivalen dalam bahasa SQL. Berdasarkan arsitektur tersebut secara prinsip penggunaannya sebagai berikut[8].

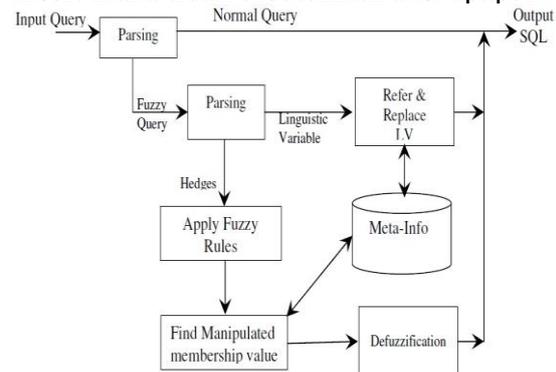
- a. Membaca setiap instruksi dari SQL query, duplikasi ke sebuah file (SQLQuery.sql) jika mengandung sebuah atribut klasik.
- b. Jika tidak, maka bagi dua atribut query fuzzy, duplikasi bagian yang memiliki atribut klasik ke SQLQuery.sql dan yang lain terjemahkan dengan FMB (Fuzzy Meta Data Knowledge Base). FMB adalah sebuah skrip yang akan menerjemahkan setiap atribut fuzzy dengan melakukan compile dengan FMB. File yang kedua berupa output yang mengandung SQL yang ekuivalen dengan fuzzy query akan dituliskan di file FMBoutput.sql.

Hasil dari langkah-langkah tersebut adalah 2 buah file yang pertama mengandung DDL yang merupakan bagian dari basis data dan yang kedua berupa file yang isinya modifikasi yang telah dilakukan oleh FMB.

Prinsip penggunaan sesuai dengan *framework* yang diusulkan oleh Balamurugan *et al* [1], *framework* tersebut dibagi menjadi 2 tahap yaitu tahap preprocessing dan tahap implementasi query. *Preprocessing* terdiri dari lima langkah. Pertama mengumpulkan informasi dari repositori yang berbeda, dibuat dalam format yang sama dan disimpan dalam suatu lokasi yang sama. Langkah kedua mengidentifikasi atribut-atribut yang dapat difuzzifikasi kemudian ketiga membentuk himpunan fuzzy yang berbeda berdasarkan atribut-atribut yang

berbeda. Langkah keempat membentuk sebuah tabel yang berisi nilai keanggotaan. Langkah *preprocessing* terakhir membentuk α -cut (α merupakan nilai atau range yang lebih rendah dari himpunan fuzzy) tabel untuk proses defuzzifikasi. Tahap implementasi query tampak pada gambar 2.

Langkah yang pertama dilakukan pengecekan terhadap query. Apakah query termasuk query seleksi atau penghapusan ataukah update, sedangkan jika query insert maka tidak dibutuhkan aksi apapun.



Gambar 2. Blok diagram implementasi query

Langkah kedua melakukan pengecekan query untuk penggunaan variabel linguistik. Jika mengandung *linguistic hedges* maka langkah berikutnya dieksekusi yaitu nilai keanggotaan dihitung dan dimanipulasi sesuai dengan *linguistic hedges*-nya. Ketika operasi ini selesai dilakukan maka sekaligus melakukan proses defuzzifikasi untuk mentransformasi variabel linguistik tersebut. Defuzzifikasi dihasilkan dengan mencari nilai α -cut dan dengan menghitung rentang maksimum dan minimum.

yang sesuai dengan data. Kemudian melakukan menerapkan perhitungan dan menguji coba query *crisp* dan query berbasis fuzzy, dibandingkan hasilnya hingga mengambil kesimpulan.

2. METODE PENELITIAN

Penelitian ini dilakukan dengan mengambil cuplikan tabel mahasiswa yang ada di basis data SIAKKE. Selanjutnya mendesain derajat keanggotaan dan fungsi keanggotaan

3. HASIL DAN PEMBAHASAN

3.1 Cuplikan Basis Data SIAKKE Offline dan Hasil Eksperimen

Dengan menggunakan cuplikan tabel mahasiswa dari basis data SIAKKE, maka pada tulisan ini diambil skema tabel mahasiswa terdiri atas NIM, SKS_diakui dan Prodi_asal. Atribut NIM adalah *primary key*. Atribut SKS_diakui adalah atribut fuzzy sedangkan atribut NIM dan Prodi_asal adalah atribut *crisp*. Tabel 1 adalah cuplikan tabel mahasiswa tersebut. Di aplikasi SIAKKE biasanya menggunakan query untuk menampilkan mahasiswa dan jumlah SKS yang diakui.

Tabel 1: Cuplikan tabel mahasiswa SIAKKE offline

NIM	SKS_diakui _i	Prodi_asal
0851049	47	D3 MI
0851050	39	D3 MI
0851051	98	D3 MI
0851052	40	S1 Elektro
0851053	24	S1 TI
0851054	72	D3 MI
0851055	57	S1 TI
0851056	36	S1 TI

Sebagai contoh menampilkan mahasiswa dengan SKS_diakui kurang dari atau sama dengan 40. Jika diterjemahkan ke dalam bahasa SQL [8] menjadi:

*Select * from mahasiswa where SKS_diakui <=40;*

Hasil dari SQL query ini seperti pada tabel 2. Dari 10 record, ada 4 buah record yang memenuhi parameter yang diminta sebelumnya.

Tabel 2: Output Sql Query

NIM	SKS_diakui	Prodi_asal
0851050	39	D3 MI
0851052	40	S1 Elektro
0851053	24	S1 TI
0851056	36	S1 TI

Pada penelitian ini menggunakan tiga buah linguistik variabel untuk merepresentasikan atribut SKS_diakui sebagai atribut fuzzy. Linguistik variabel tersebut antara lain: sedikit, rata-rata dan banyak. Interval nilai dari masing-masing variabel adalah sedikit (0-40), rata-rata (35-100) dan banyak (>90). Fungsi keanggotaan (*membership function*) dari atribut fuzzy ini sebagai berikut [8],[10]. Penggambaran derajat keanggotaan ini ke dalam bentuk fungsi keanggotaan seperti pada gambar 3. Masing-masing linguistik variabel digambarkan dalam bentuk kurva trapezoid [1].

- a. Derajat keanggotaan untuk SKS_diakui sedikit

If $(x > 0 \ \& \ x \leq 35)$ then derajat keanggotaan = 1 else $(35 < x \leq 40)$ then

$$\text{derajat keanggotaan} = \frac{(40-x)}{5}. \quad (1)$$

- b. Derajat keanggotaan untuk SKS_diakui rata-rata

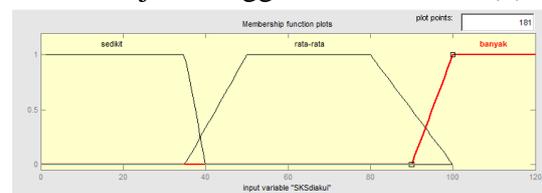
If $(x \leq 35 \ \& \ x \geq 100)$ then derajat keanggotaan = 0, else if $(35 < x < 50)$

then derajat keanggotaan = $\frac{(x-35)}{15}$, else if $(50 \leq x \leq 80)$ then derajat keanggotaan = 1, else $(80 < x \leq 100)$ then derajat keanggotaan = $\frac{(100-x)}{20}$. (2)

- c. Derajat keanggotaan untuk SKS_diakui banyak

If $(90 \leq x < 100)$ then derajat

keanggotaan = $\frac{(x-90)}{10}$, else $(x \geq 100)$ then derajat keanggotaan = 1. (3)



Gambar 3. Fungsi keanggotaan untuk variabel SKS_diakui

Plot points pada gambar sudah ditentukan secara *default* oleh software aplikasi pembuat fungsi keanggotaan ini jadi tidak ada perubahan. Rentang nilai variabel masing-masing sudah dijelaskan sebelumnya. Namun pada gambar terdapat modifikasi rentang plot hingga melebihi nilai 100 (rentang plot 0 sampai dengan 120) untuk memperlihatkan range variabel banyak agar terlihat lebih detail.

3.2 Pembahasan

Dengan menggunakan fuzzy query dapat dilakukan retrieval informasi sesuai dengan variabel linguistik yang sudah dijabarkan sebelumnya [8]. Sebagai contoh jika memanggil mahasiswa dengan SKS yang diakui yang jumlahnya sedikit. Fuzzy SQL dari contoh seperti berikut.

*Select * from mahasiswa where SKS_diakui FEQ \$sedikit;*

Hasil dari fuzzy query ini seperti tampak pada tabel 3. Query ini akan memanggil semua *record* untuk SKS_diakui sedikit serta memiliki derajat keanggotaan lebih dari 0. Dengan fuzzy query tersebut lebih mudah untuk merepresentasikan bahasa alamiah. Lebih mudah menyebutkan mahasiswa dengan SKS_diakui sedikit daripada menyebut mahasiswa dengan SKS_diakui kurang dari atau sama dengan 40. Dari hasil fuzzy query ini juga dapat diketahui kecenderungan mahasiswa berada pada variabel linguistik yang mana. Tampak pada tabel 3, seorang mahasiswa dengan SKS_diakui sebanyak 36 SKS dengan derajat keanggotaan 0.8 akan lebih besar kecenderungannya untuk berada pada variabel sedikit dibandingkan dengan mahasiswa dengan SKS_diakui 39 SKS dengan derajat keanggotaan 0.2.

Tabel 3: Output fuzzy query SKS_Diakui Sedikit

NIM	SKS_dia kui	Prodi_a sal	Derajat Keanggo tan
08510 50	39	D3 MI	0.2
08510 53	24	S1 TI	1
08510 56	36	S1 TI	0.8

Tabel 4: Output fuzzy query SKS_Diakui Sedikit dan Treshold 0.4

NIM	SKS_dia kui	Prodi_a sal	Derajat Keanggo tan
08510 53	24	S1 TI	1
08510 56	36	S1 TI	0.8

Dengan fuzzy query, ekspresi variabel linguistik dapat diperluas dengan menggunakan nilai *threshold* dari derajat keanggotaan [8]. Jika ingin melakukan *retrieve* data detail dari mahasiswa yang SKS_diakui sedikit dengan nilai *threshold* lebih dari 0.4, maka fuzzy SQL-nya:

*Select * from mahasiswa where SKS_diakui FEQ \$sedikit THOLD 0.4;*

Hasil dari fuzzy query ini tampak pada tabel 4. Query ini akan menampilkan semua *record* yang derajat keanggotaannya lebih dari 0.4 dan masih tetap berada pada variabel linguistik sedikit.

4. KESIMPULAN

Desain fuzzy query ini berhasil diaplikasikan pada tabel mahasiswa basis data SIAKKE. Dengan demikian teknik fuzzy query dapat diterapkan pada basis data relasional sehingga memudahkan user melakukan proses seleksi informasi dengan menggunakan ekspresi linguistik.

DAFTAR PUSTAKA

- [1] Balamurugan, V. and K. Senthamarai Kannan., (2010). A Framework for Computing Linguistic Hedges in Fuzzy Queries. The International Journal of Database Management Systems (IJDMS), Vol.2, No.1, Februari 2010. <http://airccse.org/journal/ijdms/papers/0210ijdms1.pdf>.
- [2] Shukla, P. K., Surya Prakash Tripathi, Shyam Kishore Bajpai., (2010). Handling Grey Information in Object Oriented Databases. International Journal of Information Technology and Knowledge Management Juli-Desember 2010, Vol.2, No.2, pp. 211-215. <http://www.csjournals.com/IJITKM/PDF%203-1/7.pdf>.
- [3] Deepa S., (2014). An Approach for Normalizing Fuzzy Relational Databases Based on Join Dependency. International Journal of Computer Trends and Technology (IJCTT) V9(1):1-3, Maret 2014. ISSN:2231-2803. <http://arxiv.org/ftp/arxiv/papers/1403/1403.2958.pdf>
- [4] Shukla, P.K., Manuj Darbari, Vivek Kumar Singh, Surya Prakash Tripathi. (2011). A Survey of Fuzzy Techniques in Object Oriented Databases. International Journal of Scientific & Engineering Research, Volume 2, Issue 11, November-2011, ISSN 2229-5518. <http://www.ijser.org/researchpaper/A-Survey-of-Fuzzy-Techniques-in-Object-Oriented-Databases.pdf>
- [5] Vucetic, M. (2011). Functional Dependencies Analyse in Fuzzy Relation Database Models. Jurnal of Information Technology and Information. Vol.1, No. 2. Desember 2011, [http://www.jita-](http://www.jita-au.com/Public/PDF/JITA_Vol%201_Issue2.pdf)
- [6] Singh, S., Kavita Agarwal, Jameel Ahmad. (2014). Conceptual Modelling in Fuzzy Object-Oriented Databases Using Unified Modelling Language. International Journal of Latest Research in Science and Technology, Vol.3, Issue 3, pp. 174-178, Mei-Juni 2014. ISSN (online): 2278-5299. http://www.mnkjournals.com/ijlrst_files/download/Vol%203%20Issue%203/32-76-25062014%20Conceptual%20modeling%20in%20Fuzzy%20Object-Oriented%20Databases%20using%20Unified%20Modeling%20Language.pdf.
- [7] Irwan, D. (2008). Implementasi Fuzzy Query pada Database untuk Rekomendasi Beasiswa. Sarjana, Universitas Sumatra Utara. http://repository.usu.ac.id/bitstream/123456789/11888/1/09E00028.pdf?origin=publication_detail.
- [8] Kumari S., Sonia. (2013). Fuzzy RDBMS Design: SQL Add-on. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET). Vol.2, No. 5, Mei 2013. ISSN: 2278-1323. <http://ijarcet.org/wp-content/uploads/2013/06/1861-1865.pdf>.
- [9] Gupta, P. (2011). Database Design for Storage of Fuzzy Information in Traditional Database. International Journal of Computer Applications. Vol. 15, No. 2, Februari 2011. ISSN: 0975-8887. <http://www.ijcaonline.org/archives/volume15/number2/1916-1554>.
- [10] Mondal, S.K., Joyassree Sen, Md. Rabiul Islam, Md. Shamim Hossian.(2013). Performance Comparison of Fuzzy Queris on Fuzzy Database and Classical

Database. International Journal of
Computer Application. Issue 3,
Vol.1, Februari 2013. ISSN: 2250-
1797.

http://www.rpublication.com/ijca/ijca_index.htm.