

IMPLEMENTASI *FUZZY* OLAP PADA DATA POTENSI DESA DI PROVINSI JAWA BARAT TAHUN 2003 DAN 2006

Imas S Sitanggang¹, Sri Nurdiati¹, Sofiyanti Indriasari²

¹ Staf Departemen Ilmu Komputer, Fakultas Matematika dan IPA, Institut Pertanian Bogor

² Mahasiswa Departemen Ilmu Komputer, Fakultas Matematika dan IPA, Institut Pertanian Bogor

ABSTRAK

Kurangnya pemanfaatan atau pengolahan terhadap gunung data yang sebenarnya dapat menghasilkan suatu informasi atau pengetahuan yang penting untuk mendukung pengambilan keputusan. Data Potensi Desa yang bersumber dari Badan Pusat Statistik (BPS) memiliki peranan yang cukup strategis di masa yang akan datang dan merupakan produk unggulan BPS. Untuk itu sangat penting melakukan analisis data Podes secara efisien dan cepat. *Data warehouse* adalah tempat penyimpanan data terintegrasi yang dapat digunakan untuk *query* dan analisis. Operasi *query* ini dilakukan dengan *On-line Analytical Processing* (OLAP). Data real dijumpai sering kali mengandung informasi yang tidak tepat dan ketidakpastian (*uncertain*). Teori himpunan *fuzzy* digunakan untuk menangani masalah tersebut. Penelitian ini bertujuan membangun *data warehouse* untuk data Potensi Desa di Provinsi Jawa Barat tahun 2003 dan 2006 yang direpresentasikan dalam model data multidimensi dengan menggunakan teori himpunan *fuzzy* dan mengimplementasikan operasi OLAP.

Ruang lingkup penelitian adalah dibatasi pada implementasi *fuzzy* OLAP untuk data Potensi Desa di Provinsi Jawa Barat tahun 2003 dan 2006. Sistem yang akan dibangun mengikuti arsitektur *data warehouse* tiga tingkat. Struktur data pada *data warehouse* digambarkan dengan skema galaksi. Operasi-operasi yang diimplementasikan adalah operasi *roll-up*, *drill-down*, *slice*, *dice*, dan *pivot*. Operasi untuk *fuzzy* OLAP dilakukan sesuai dengan konsep pengoperasian data multidimensi *fuzzy*.

Hasil penelitian ini adalah terbentuknya *data warehouse* yang diimplementasikan dengan konsep *fuzzy* serta *data warehouse* untuk data *crisp* sebagai pelengkap penyajian informasi. *Data warehouse* yang dibangun menggunakan konsep *fuzzy* terlebih dahulu melalui tahapan pra-proses data dan proses *clustering* menggunakan algoritma FCM untuk mendapatkan himpunan *fuzzy* dan nilai derajat keanggotaan tiap atribut. *Data warehouse* tersebut memiliki 4 kubus data yaitu penduduk, rumah tangga, lahan, dan sekolah. Data dapat ditampilkan dalam bentuk *crossstab* atau grafik. Penelitian ini memberikan informasi berupa data *fuzzy* dan data *crisp*. Data *fuzzy* yang disajikan memberikan informasi natural dengan mendefinisikan variabel *linguistic*. Data *crisp* disajikan sebagai penunjang untuk mendapatkan informasi.

Kata kunci: *data warehouse*, OLAP, model data multidimensi, *fuzzy*.

PENDAHULUAN

Latar Belakang

Dewasa ini, ketersediaan data semakin melimpah, apalagi ditunjang dengan banyaknya kegiatan yang sudah dilakukan secara terkomputerisasi. Namun seringkali data tersebut hanya disimpan tanpa diolah lebih lanjut untuk keperluan di masa mendatang. Padahal jika dianalisis lebih dalam, data tersebut dapat menghasilkan informasi atau pengetahuan yang penting dan berharga. Data Potensi Desa (Podes) yang bersumber dari Badan Pusat Statistik (BPS) memiliki peranan yang cukup strategis di masa

yang akan datang dan merupakan produk unggulan BPS. Untuk itu sangat penting dilakukan analisis data Podes secara efisien dan cepat.

Data warehouse adalah tempat penyimpanan data terintegrasi yang dapat digunakan untuk *query* dan analisis. Operasi *query* ini dilakukan dengan *On-line Analytical Processing* (OLAP). OLAP akan menghasilkan data dalam bentuk ringkasan. Dengan operasi ini *query* dapat dijalankan lebih mudah dan lebih efisien. OLAP muncul dengan sebuah cara pandang data multidimensi. Cara pandang multidimensi ini didukung oleh teknologi basis data

multidimensi. *Data warehouse* dan *On-Line Analytical Processing* (OLAP) merupakan elemen penting dalam mendukung proses pengambilan keputusan.

Data real yang dijumpai sering kali mengandung informasi yang tidak tepat dan ketidakpastian (*uncertain*). Teori himpunan *fuzzy* digunakan untuk menangani masalah tersebut.

Tujuan Penelitian

Tujuan dari penelitian ini adalah :

1. Membangun *data warehouse* untuk data Potensi Desa tahun 2003 dan 2006. Aplikasi yang dibangun mengolah data yang direpresentasikan dalam model data multidimensi dengan menggunakan teori himpunan *fuzzy*.
2. Mengimplementasikan operasi-operasi OLAP, yaitu *roll-up*, *drill-down*, *slice*, *dice*, dan *pivot* dengan menggunakan pendekatan *fuzzy*.

Ruang Lingkup Penelitian

Penelitian ini dibatasi pada implementasi *fuzzy* OLAP pada data Potensi Desa di Provinsi Jawa Barat tahun 2003 dan 2006 yang bersumber dari Badan Pusat Statistik (BPS). Penelitian ini akan menghasilkan aplikasi *fuzzy* OLAP yang dapat menampilkan informasi untuk mendukung proses pengambilan keputusan.

Manfaat Penelitian

Aplikasi yang dihasilkan pada penelitian ini diharapkan dapat digunakan oleh pihak yang berkepentingan untuk mendapatkan informasi yang menarik dan akurat sebagai pendukung proses pengambilan keputusan terkait dengan data Potensi Desa. Sistem diharapkan dapat menyediakan informasi ringkas dan jelas secara cepat melalui operasi-operasi OLAP seperti *roll-up*, *drill-down*, *slice*, *dice*, dan *pivot*.

TINJAUAN PUSTAKA

Data Warehouse

Data Warehouse adalah sekumpulan data berorientasi subjek, terintegrasi, *time-variant*, dan *non-volatile* yang mendukung proses manajemen pembuatan keputusan (Inmon 1996).

Model Data Multidimensi

Model data multidimensi menampilkan data dalam bentuk kubus data. Kubus data memungkinkan data dimodelkan dan ditampilkan dalam dimensi banyak. Kubus data disebut juga *cuboid*. Pola-pola *cuboid* dapat dibuat apabila diberikan satu kumpulan dimensi. Masing-masing pola menampilkan data pada tingkat kesimpulan yang berbeda-beda (Han & Kamber 2001).

Untuk menggambarkan hubungan antardata pada data multidimensi digunakan skema multidimensi. Skema adalah sekumpulan objek basis data. Pada *data warehouse*, skema merupakan sekumpulan tabel yang berhubungan. Skema digunakan untuk menunjukkan hubungan antara tabel dimensi dengan tabel fakta. Skema ditentukan berdasarkan kebutuhan *data warehouse* dan keinginan pembuat *data warehouse*. *Data warehouse* membutuhkan skema yang ringkas dan berorientasi subjek. Tipe-tipe skema multidimensi antara lain (Han & Kamber 2001):

- Skema bintang (*star schema*)

Skema ini disebut skema bintang karena hubungan antara tabel dimensi dan tabel fakta menyerupai bintang di mana satu tabel fakta dihubungkan dengan beberapa tabel dimensi. Titik tengah skema bintang adalah satu tabel fakta besar dan sudut-sudutnya adalah tabel-tabel dimensi.

- Skema *snowflake* (*snowflake schema*)

Skema *snowflake* adalah variasi dari skema bintang dimana beberapa tabel dimensi dinormalisasi, jadi dihasilkan beberapa tabel tambahan.

- Skema galaksi (*fact constellation*)

Pada skema galaksi, beberapa tabel fakta berbagi tabel dimensi.

On-line Analytical Processing (OLAP)

OLAP adalah operasi basis data untuk mendapatkan data dalam bentuk ringkasan dengan menggunakan agregasi sebagai mekanisme utama. OLAP mengerjakan *query-query* rumit, tapi hanya satu yang dikerjakan dalam satu waktu. *Query* yang dihasilkan OLAP tidak mengubah data.

Tipe-tipe operasi OLAP antara lain (Han & Kamber 2001):

- *Roll-up*

Operasi ini melakukan pengelompokan pada kubus data dengan cara mengurutkan suatu hirarki konsep secara menaik (*ascending*) atau pengurangan dimensi.

Contoh :

Suatu hirarki lokasi: kota < provinsi < negara

Operasi *roll-up* melakukan pengelompokan dengan mengurutkan hirarki lokasi dari tingkat kota sampai pada tingkat negara.

- *Drill-down*

Drill-down adalah kebalikan dari *roll-up*. *Drill-down* dapat dilakukan dengan cara mengurutkan suatu hirarki konsep secara menurun (*descending*) atau dengan menambahkan nilai dimensi.

Contoh :

Suatu hirarki waktu : bulan < kuartal < tahun

Drill-down melakukan pengelompokan dengan mengurutkan hirarki waktu dari tingkat tahun ke tingkat yang lebih detil yaitu tingkat bulan.

- *Slice dan dice*

Operasi *slice* melakukan pemilihan satu dimensi dari kubus data sehingga menghasilkan bagian kubus (*subcube*). Operasi *dice* menghasilkan bagian kubus (*subcube*) dengan melakukan pemilihan dua atau lebih dimensi.

- *Pivot (rotate)*

Pivot adalah operasi yang memutar koordinat data pada tampilan dengan tujuan untuk menyediakan alternatif tampilan data.

Himpunan Fuzzy

Konsep logika *fuzzy* pertama kali diperkenalkan oleh Prof. Lotfi A Zadeh dari Universitas California pada bulan Juni 1965. Logika *fuzzy* merupakan generalisasi dari logika klasik yang hanya memiliki dua nilai keanggotaan 0 dan 1. Dalam logika *fuzzy* nilai kebenaran suatu pernyataan berkisar dari sepenuhnya benar ke sepenuhnya salah. Inti dari himpunan *fuzzy* yaitu fungsi keanggotaan yang menggambarkan hubungan antara

domain himpunan *fuzzy* dengan nilai derajat keanggotaan. Derajat keanggotaan menunjukkan nilai keanggotaan suatu objek pada suatu himpunan. Nilai keanggotaan ini berkisar antara 0 sampai 1 (Cox 2005). Teori himpunan *fuzzy* telah banyak digunakan dalam pembangunan sistem *data mining*. Alhaji dan Mehmet (2003) telah menggunakan himpunan *fuzzy* untuk membangun kubus data *fuzzy*.

Fuzzy C-Means (FCM)

Menurut Jang *et al.* (1997), *Fuzzy C-Means* merupakan algoritma *clustering* data di mana setiap titik data masuk dalam sebuah *cluster* dengan ditandai oleh derajat keanggotaan. FCM membagi sebuah koleksi dari n data vektor x_j ($j=1, 2, \dots, n$) menjadi c *cluster*, dan menemukan sebuah pusat *cluster* (*center*) untuk tiap kelompok dengan meminimalisasi ukuran dari fungsi objektif. Pada FCM hasil dari *clustering* adalah sebuah titik data dapat menjadi anggota untuk beberapa *cluster* yang ditandai oleh derajat keanggotaannya antara 0 dan 1.

Berikut tahapan *clustering* menggunakan algoritma FCM:

1. Inisialisasi keanggotaan matriks U yang berisi derajat keanggotaan terhadap *cluster* dengan nilai antara 0 dan 1, sehingga

$$\sum_{i=1}^c u_{ij} = 1, \quad \forall j = 1, \dots, n$$

2. Penghitungan c sebagai pusat *cluster*, c_i , $i = 1, \dots, c$ dengan menggunakan

$$c_i = \frac{\sum_{j=1}^n ((u_{ij})^m x_j)}{\sum_{j=1}^n (u_{ij})^m}$$

3. Penghitungan fungsi objektif (J_i):

$$J(U, c_1, \dots, c_c) = \sum_{i=1}^c J_i = \sum_{i=1}^c \sum_{j=1}^n u_{ij}^m d_{ij}^2$$

di mana:

- u_{ij} adalah elemen matriks U yang bernilai antara 0 dan 1,
- $d_{ij} = \|c_i - x_j\|$ adalah jarak antara pusat *cluster* ke- i dan titik data ke- j ,
- c_i adalah pusat *cluster* ke- i ,
- $m \in [1, \infty]$ adalah parameter *fuzzifikasi*. Nilai m yang umum digunakan untuk *clustering* data *fuzzy* adalah sama dengan 2.

Kemudian melihat kondisi berhenti :

- Jika $(|J_t - J_{t-1}| < \text{nilai toleransi terkecil yang diharapkan})$ atau ($t > \text{maksimal iterasi}$) maka proses berhenti.
 - Jika tidak : $t = t + 1$, mengulangi langkah 3.
4. Penghitungan matriks U baru menggunakan formula berikut:

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{d_{ij}}{d_{kj}} \right)^{\frac{2}{(m-1)}}$$

Ukuran Kevalidan Cluster

Menurut Xie dan Beni (1991), ukuran kevalidan *cluster* merupakan proses evaluasi hasil *clustering* untuk menentukan *cluster* mana yang terbaik. Kevalidan sebuah *cluster* (S) ditentukan oleh dua hal yaitu: ukuran kedekatan antaranggota pada tiap *cluster* (*compactness*), dan ukuran keterpisahan antarcluster satu dengan *cluster* yang lainnya (*separation*). Semakin kecil nilai S, maka *cluster* tersebut semakin valid.

$$S = \frac{\sum_{i=1}^c \sum_{j=1}^n \mu_{ij}^2 \|V_i - X_j\|^2}{n \min_{i,j} \|V_i - V_j\|^2}$$

dengan:

- $\frac{\sum_{i=1}^c \sum_{j=1}^n \mu_{ij}^2 \|V_i - X_j\|^2}{n}$ adalah nilai *compactness*,
- n adalah banyaknya titik data,
- V adalah pusat *cluster*,
- $\sum_{j=1}^n d_{ij}^2$ adalah *variation* dari *cluster* ke-i,
- $d_{ij} = \mu_{ij} \|X_j - V_i\|$ adalah *fuzzy deviation* X_j dari *cluster* ke-i, dan notasi $\|\bullet\|$ biasanya merupakan norma (panjang) penghitungan jarak *Euclidean*,
- d_{ij} adalah jarak antara X_j dan V_i yang diboboti oleh derajat keanggotaan *fuzzy* titik ke-j pada *cluster* ke-i,

- $\min_{i,j} \|V_i - V_j\|^2$ adalah *separation* yang merupakan jarak minimum antarpusat *cluster*.

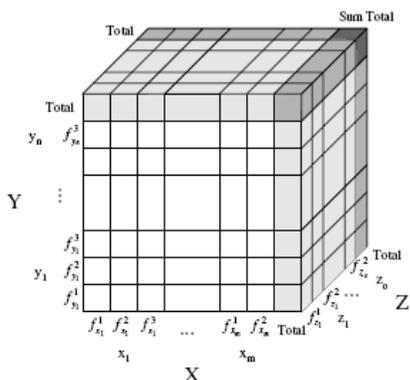
Kubus Data Fuzzy

Himpunan *fuzzy* dapat didefinisikan untuk atribut kuantitatif, misalkan x, dengan fungsi keanggotaan setiap himpunan *fuzzy* sedemikian sehingga nilai dari x memenuhi persyaratan berada dalam satu atau lebih himpunan-himpunan *fuzzy* tersebut. Misalkan $F_x = \{f_x^1, f_x^2, \dots, f_x^n\}$ adalah himpunan dari n himpunan *fuzzy* untuk atribut x. Fungsi keanggotaan dari himpunan *fuzzy* ke-j dalam F_x , dinotasikan dengan $\mu_{f_x^j}$, menyatakan

pemetaan dari domain untuk x ke dalam interval [0, 1]. Jika $\mu_{f_x^j}(v) = 1$ maka nilai v dari x secara penuh merupakan anggota untuk himpunan *fuzzy* f_x^j . Jika $\mu_{f_x^j}(v) = 0$ berarti

bahwa v bukanlah anggota dari f_x^j . Semua nilai yang lain di antara 0 dan 1, menentukan keanggotaan parsial (Alhaji & Mehmet 2003). Konsep tersebut selanjutnya digunakan untuk membangun kubus data *fuzzy*.

Gambar 1 menunjukkan kubus data *fuzzy* 3 dimensi, di mana setiap dimensi memiliki dua atribut dan banyaknya fungsi keanggotaan dari setiap atribut berkisar 2 dan 3. Dalam Gambar 1, setiap dimensi dari kubus data mengandung $\sum_{i=1}^k n_i + 1$ nilai dengan n_i adalah banyaknya fungsi keanggotaan dari atribut x_i , k adalah banyaknya atribut, keduanya dalam dimensi X, dan “+1” menunjukkan nilai total di mana setiap sel menyimpan nilai agregasi dari baris-baris sebelumnya. Seperti halnya dalam kubus data *crisp*, terhadap kubus data *fuzzy* dapat diaplikasikan operasi-operasi seperti *dice*, *slice*, *roll-up* dan *drill-down*.



Gambar 1 Contoh kubus data *fuzzy* 3 dimensi (Alhajj & Mehmet 2003)

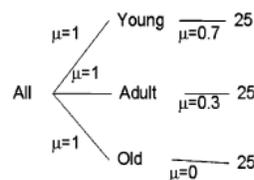
Dimensi Fuzzy

Elemen (sel) pada dimensi data *crisp* dapat digabungkan menjadi satu elemen pada level di atasnya. Untuk kasus *fuzzy*, suatu elemen dapat direlasikan dengan lebih dari satu elemen pada level di atasnya dan derajat dari relasi tersebut berada pada interval [0,1]. *Kinship relation* mendefinisikan derajat dari relasi tersebut sebagai berikut (Molina *et al.* 2006):

Untuk tiap pasangan level l_i dan l_j di mana $l_j \in H_i$, didefinisikan

$$\mu_{ij} : l_i \times l_j \rightarrow [0,1]$$

Setiap derajat dari elemen-elemen dimasukkan di dalam satu elemen pada level *parent* dapat dilakukan dengan menggunakan relasi tersebut. Dengan menggunakan relasi antarelelemen dalam dua level berurutan, dapat didefinisikan relasi antara tiap-tiap pasangan nilai dalam level yang berbeda di suatu dimensi yang disebut *extended kinship relation*. Untuk menghasilkan suatu nilai agregasi, dilakukan pertimbangan terhadap semua kemungkinan *path* antarelelemen dalam hirarki. Masing-masing nilai dikalkulasikan dengan mengagregasikan *kinship relation* pada dua level menggunakan operator \otimes dan \oplus di mana operator tersebut secara berurutan adalah *t-norm* dan *t-conorm* yaitu implementasi dari operator minimum dan maksimum. Contoh sederhana dapat dilihat pada Gambar 2 (Molina *et al.* 2006).



Gambar 2 Contoh *path* pada dimensi *fuzzy*

Gambar 2 menunjukkan *path* yang akan menjadi alur untuk mengkalkulasikan nilai agregasi. Proses kalkulasi nilai-nilai dari contoh tersebut adalah sebagai berikut:

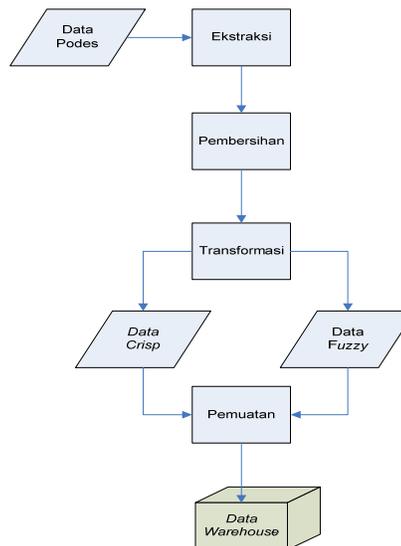
$$(1 \otimes 0.7) \oplus (1 \otimes 0.3) \oplus (1 \otimes 0) = 0.7$$

METODE PENELITIAN

Tahapan Penelitian

Tahapan penelitian terdiri dari empat tahapan yaitu:

1. Pengumpulan dataset yang terdiri dari atribut numerik
2. Tahapan praproses data meliputi ekstraksi, pembersihan, transformasi, dan pemuatan (*load*). Tahap-tahap praproses data dapat dilihat pada Gambar 3.



Gambar 3 Tahap praproses data

3. Perancangan dan konstruksi data multidimensi *fuzzy*
4. Perancangan antarmuka kueri *Fuzzy OLAP*.

Praproses Data

Praproses data adalah proses yang harus dilakukan sebelum membuat *data warehouse*. Proses-proses tersebut adalah:

1 Ekstraksi (*extraction*)

Ekstraksi adalah pengambilan data yang relevan untuk analisis dari basis data operasional sebelum masuk ke *data warehouse*. Pada ekstraksi, atribut-atribut dan *record-record* yang diinginkan dipilih dan diambil dari basis data operasional. Hal ini perlu dilakukan karena tidak semua elemen data berguna dalam analisis dan pembuatan keputusan. Elemen data yang diperlukan adalah atribut pada data Podes Jawa Barat tahun 2003 dan 2006 yang bersifat numerik agar dapat diimplementasikan konsep *fuzzy* dan merupakan atribut data yang mengandung konsep hirarki agar dapat terlihat pola agregasi pada saat pengimplementasian operasi OLAP, serta atribut yang terdapat pada data Potensi Desa Jawa Barat tahun 2003 dan 2006 antartahun.

2 Pembersihan (*cleaning*)

Pada pembersihan semua kesalahan dihilangkan dan diperbaiki. Pembersihan dilakukan untuk menghilangkan kesalahan (*error*). Kesalahan yang umum terjadi adalah nilai yang hilang (*missing values*), *noise*, dan data yang tidak konsisten. Pembersihan dilakukan dengan mengisi nilai yang kosong dan menghilangkan *noise*.

3 Transformasi (*transformation*)

Pada transformasi, data Podes Jawa Barat tahun 2003 dan 2006 yang terpilih diberikan format dan nama yang umum. Proses transformasi dilakukan agar data tetap konsisten. Transformasi yang paling penting adalah transformasi nama agar tidak ada nama atribut yang sama atau atribut yang sama memiliki nama yang berbeda pada basis data yang berbeda. Setelah transformasi nama, semua elemen data harus dikonversikan ke format yang sama.

Transformasi selanjutnya adalah transformasi nilai dari data *crisp* ke data *fuzzy*. Pembentukan data *fuzzy* dilakukan melalui proses *clustering* menggunakan algoritma FCM untuk menentukan

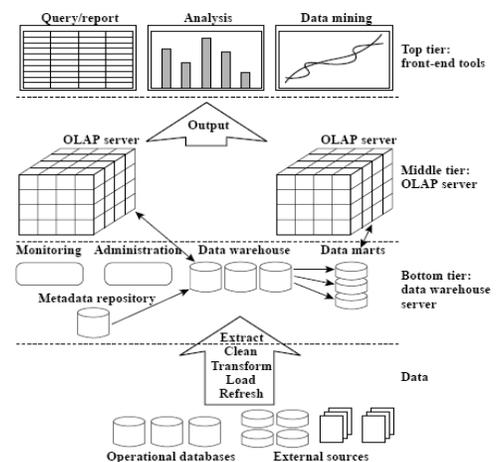
himpunan *fuzzy* dan derajat keanggotaannya. Langkah selanjutnya adalah menentukan validitas *cluster*. Tujuan dilakukannya validasi terhadap *fuzzy clustering* yaitu untuk mencari skema *clustering* di mana sebagian besar vektor dari suatu himpunan data menunjukkan derajat keanggotaan yang tinggi dalam suatu *cluster*. Pengukuran kevalidan *cluster* dilakukan menurut metode Xie dan Beni (1991) yang menjelaskan bahwa kevalidan sebuah *cluster* (S) ditentukan oleh dua hal yaitu: ukuran kedekatan antaranggota pada tiap *cluster* (*compactness*), dan ukuran keterpisahan antar*cluster* satu dengan *cluster* yang lainnya (*separation*). Semakin kecil nilai S, maka *cluster* tersebut semakin valid.

4 Pemuatan (*loading*)

Setelah tahap ekstraksi, pembersihan, dan transformasi dilakukan, maka data dapat dimasukkan ke *data warehouse*. Data yang dimasukkan ke dalam *data warehouse* adalah data *crisp* dan data *fuzzy*.

Arsitektur Sistem

Sistem yang akan dibangun mengikuti arsitektur *data warehouse* tiga tingkat seperti terlihat pada Gambar 4:



Gambar 4 Arsitektur tiga tingkat *data warehouse* (Han & Kamber 2001)

1 Tingkat bawah (*bottom tier*)

Tingkat bawah arsitektur *data warehouse* adalah *server* basis *data warehouse* yang biasanya sebuah sistem basis data relasional. Tahap ini data

diambil dari basis data operasional dan sumber eksternal lainnya. Data disimpan sebagai *data warehouse*.

2. Tingkat tengah (*middle tier*)

Pada tingkat tengah, operasi OLAP dilakukan pada *data warehouse* yang sudah terbentuk. Implementasi operasi OLAP disesuaikan dengan struktur data Podes Jawa Barat 2003 dan 2006, yang mencakup struktur hirarki data. Pada penelitian ini, operasi OLAP diimplementasikan sesuai dengan aturan-aturan pada konsep *fuzzy*, termasuk fungsi agregasi *fuzzy* yang digunakan.

3. Tingkat atas (*top tier*)

Pada tingkat atas, *data warehouse* telah terbentuk dan digunakan oleh pengguna. Perangkat analisis berupa perangkat *query* dan grafik dibuat untuk memudahkan analisis data.

Lingkungan Pengembangan Sistem

Spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam pengembangan sistem adalah sebagai berikut:

- a. Perangkat keras berupa komputer personal dengan spesifikasi:
 - *Processor*: AMD Athlon™ XP 2700+
 - *Memory*: 768 MB
 - *Harddisk* 40 GB
 - *Keyboard* dan *mouse*
 - Monitor
- b. Perangkat lunak:
 - Sistem operasi: Microsoft Windows XP Professional
 - DBMS: Oracle 10g *Release 2*
 - *Data Warehouse* dan OLAP: Analytic Workspace Manager dan Measure Data Viewer
 - MATLAB 7 untuk *clustering* data
 - Microsoft SQL Server 2000 dan Microsoft Excel untuk praproses data.

HASIL DAN PEMBAHASAN

Praproses Data

Praproses dilakukan terhadap data Podes Jawa Barat 2003 dan 2006, tahapan praproses yang dilakukan adalah sebagai berikut:

1. Ekstraksi

Tahapan ekstraksi adalah melakukan pemilihan atribut yang relevan untuk analisis yaitu atribut numerik, atribut berhirarki, dan atribut yang ada pada data Potensi Desa Jawa Barat tahun 2003 dan 2006 antartahun. Atribut pada data Potensi Desa Jawa Barat tahun 2003 dan 2006 yang terpilih adalah atribut lokasi meliputi desa, kecamatan, dan kabupaten di Jawa Barat, serta atribut yang berhubungan dengan kependudukan yang meliputi jumlah penduduk dan jumlah rumah tangga prasejahtera I serta jumlah keluarga keseluruhan di masing-masing desa, luasan lahan, dan pendidikan yang meliputi jumlah sekolah dari TK sampai SMA.

2. Pembersihan

Tahap pembersihan data tidak perlu dilakukan, karena data yang digunakan sudah bersih.

3. Transformasi

Tahap transformasi adalah memberikan format yang umum pada data, meliputi menyamakan nama atribut yang sama pada data tahun 2003 dan 2006 atau memberikan nama atribut dengan nama yang umum sehingga atribut antartahun menjadi seragam dan lebih informatif. Selain itu, juga dilakukan penggabungan atribut pada data tahun 2003 karena pada data tahun 2006 beberapa atribut yang sesuai dengan atribut data tahun 2003 dijadikan satu. Pembentukan atribut baru juga dilakukan yang bertujuan untuk mendefinisikan secara lengkap hirarki data. Atribut baru yang dibuat adalah atribut "Keluarga bukan Prasejahtera I". Nilai-nilai atribut tersebut didapatkan dari mengurangi nilai atribut jumlah keluarga keseluruhan dengan nilai atribut jumlah keluarga Prasejahtera I, sehingga nantinya akan dapat didefinisikan hirarki data untuk dimensi rumah tangga.

Pada penelitian kali ini, data yang akan dimasukkan ke dalam *data warehouse* bukan hanya data *fuzzy* tetapi juga data dalam bentuk *crisp*. Praproses data yang telah dilakukan sebelumnya adalah menghasilkan data *crisp*, untuk itu perlu adanya proses lanjutan untuk menghasilkan data dalam bentuk *fuzzy*. Pembentukan data

fuzzy dilakukan dengan proses *clustering* terhadap seluruh atribut yang telah terpilih dengan menggunakan algoritma FCM untuk menentukan himpunan *fuzzy* dan fungsi keanggotaannya.

Pada proses *clustering* menggunakan algoritma FCM, dapat menentukan jumlah *cluster* yang diinginkan. Akan tetapi, untuk memilih *cluster* yang terbaik dilakukan pengukuran kevalidan *cluster*, sehingga proses *clustering* dicobakan berulang-ulang dengan jumlah *cluster* yang berbeda-beda kemudian dilihat nilai kevalidan *cluster*. *Cluster* yang valid adalah yang memiliki nilai kevalidan *cluster* (S) terkecil.

Pada penelitian kali ini, penentuan himpunan *fuzzy* tidak hanya berdasarkan pada ukuran kevalidan *cluster* tetapi juga memperhatikan kesamaan jumlah *cluster* pada dimensi yang sama. Hal ini dilakukan dengan pertimbangan bahwa dalam struktur *data warehouse* atribut dalam satu dimensi yang sama harus sama jumlahnya termasuk *measure* yang digunakan juga harus sama.

Clustering dilakukan terpisah antara atribut-atribut dalam data tahun 2003 dan 2006, sehingga memungkinkan terjadinya ketidaksamaan jumlah *cluster* yang valid antartahun pada dimensi yang sama. Jika terjadi ketidaksesuaian maka dilakukan penyesuaian dengan memilih selisih nilai S terkecil. Atribut yang memiliki selisih S terkecil akan dipilih *cluster* yang jumlahnya sama dengan atribut yang berada pada dimensi yang sama dan memiliki selisih nilai S lebih besar. Jadi pemilihan *cluster* pada atribut tersebut bukan berdasarkan pada nilai S terkecilnya, tapi disesuaikan dengan nilai S terkecil atribut lain dalam satu dimensi. Selain itu, percobaan untuk menentukan jumlah *cluster* hanya dibatasi sampai lima *cluster* saja, hal ini disebabkan karena jika lebih dari lima *cluster* yang terbentuk maka variabel *linguistic* yang dihasilkan menjadi kurang informatif.

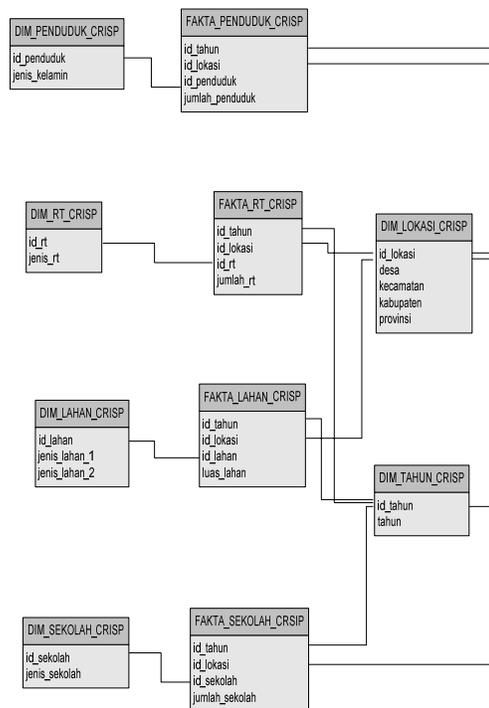
4. Pemuatan (*loading*)

Tahap pemuatan atau *loading* adalah tahap pemasukan data yang sudah siap ke *data warehouse*.

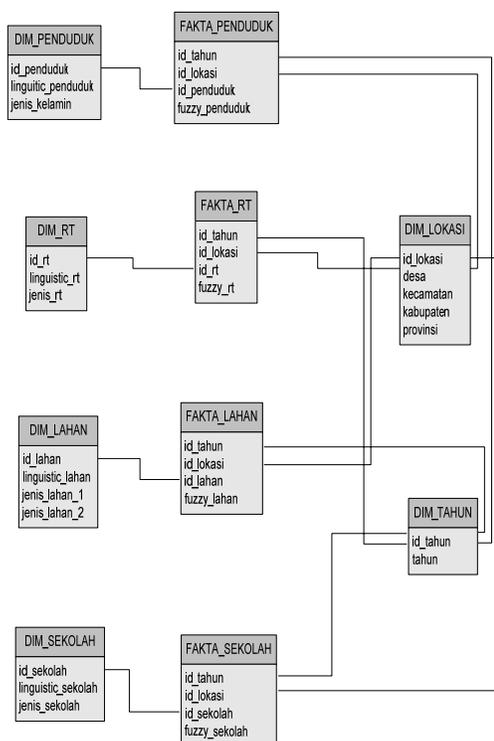
Tingkat Bawah (*Bottom Tier*)

Pada tingkat ini data diambil dari basis data operasional. Selain itu, dilakukan penyesuaian struktur data terhadap skema *data warehouse*. Data yang dimasukkan ke dalam *data warehouse* disesuaikan dengan rancangan skema data multidimensi yang diinginkan, untuk menggambarkan hubungan antardata. Sebelum skema didefinisikan, terlebih dahulu dilakukan pembuatan tabel-tabel yang nantinya akan digunakan untuk mendefinisikan hubungan antardata yang digambarkan pada skema.

Setelah itu, dilakukan perancangan skema yang digunakan untuk menunjukkan hubungan antara tabel dimensi dengan tabel fakta. Skema ditentukan berdasarkan kebutuhan *data warehouse* dan keinginan pembuat *data warehouse*. Penelitian ini menggunakan skema galaksi untuk menggambarkan hubungan antardata. Skema galaksi untuk data *crisp* yang terbentuk dapat dilihat pada Gambar 5 dan skema galaksi untuk data *fuzzy* terlihat pada Gambar 6.



Gambar 5 Skema galaksi untuk data *crisp*



Gambar 6 Skema galaksi untuk data *fuzzy*

Tingkat Tengah (*Middle Tier*)

Pada tingkat tengah adalah implementasi operasi OLAP pada *data warehouse* yang sudah terbentuk. Operasi OLAP yang diimplementasikan meliputi operasi *roll-up*, *drill-down*, *slice*, *dice*, dan *pivot*. Pada penelitian ini, operasi OLAP pada data *crisp* menggunakan fungsi agregasi biasa yaitu menggunakan operator agregasi berupa *sum*. Untuk data *fuzzy* operasi OLAP diimplementasikan sesuai dengan aturan-aturan pada konsep *fuzzy*. Untuk menghasilkan suatu nilai agregasi, dilakukan pertimbangan terhadap semua kemungkinan *path* antarelemen dalam hirarki. Masing-masing nilai dikalkulasikan dengan mengagregasikan *kinship relation* menggunakan operator \otimes dan \oplus di mana operator tersebut secara berurutan adalah *t-norm* dan *t-conorm*. Nilai pada level yang saling berurutan di suatu *path* dikalkulasikan dengan menggunakan operator *t-norm* kemudian hasil dari kalkulasi tiap-tiap *path* diagregasikan menggunakan operator *t-conorm* untuk mendapatkan nilai agregasi pada level yang lebih tinggi. Fungsi *t-norm* adalah mengambil nilai minimum dari *path* yang levelnya berurutan, sedangkan fungsi *t-*

conorm adalah mengambil nilai maksimum dari hasil kalkulasi pada tiap-tiap *path* yang telah dikalkulasikan dengan fungsi *t-norm* sebelumnya.

Tingkat Atas (*Top Tier*)

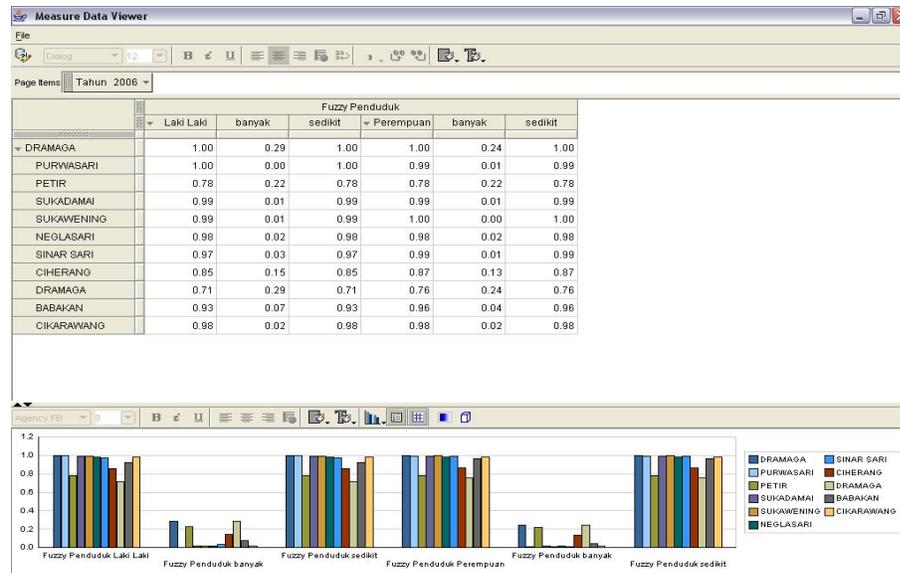
Pada tingkat *top tier* adalah menampilkan *data warehouse* yang telah terbentuk, sehingga dapat digunakan oleh pengguna. Penyajian data dapat dilakukan dari berbagai sudut pandang, misalnya ringkasan data yang menampilkan elemen-elemen yang merupakan potensi daerah di Provinsi Jawa Barat atau penyajian data secara detil di elemen terkecil yaitu tingkat desa. Tampilan yang terbentuk juga dilengkapi dengan grafik sebagai bentuk visualisasi data. Selain itu, juga dilengkapi dengan *query builder* dalam bentuk *query wizard* yang dapat mempermudah pengguna untuk menjalankan operasi OLAP.

Nilai-nilai yang ditampilkan di setiap elemen pada kubus data *crisp* adalah berupa nilai numerik yang sesuai dengan sumber data aslinya.

Nilai-nilai yang ditampilkan pada kubus data *fuzzy* adalah nilai derajat keanggotaan terhadap himpunan *fuzzy* yang telah dibentuk dan diproses terlebih dahulu. Nilai derajat keanggotaan pada level lebih tinggi merupakan nilai maksimum dari level di bawahnya. Untuk memahami makna nilai agregasi tersebut adalah dengan mempertimbangkan perbandingan nilai derajat keanggotaan antarlinguisticnya, misalnya jika derajat keanggotaan pada *linguistic A* lebih besar daripada nilai derajat keanggotaan pada *linguistic B* maka nilai tersebut masuk ke dalam kategori A. Akan tetapi, jika nilai hasil agregasi menunjukkan bahwa perbandingan nilai *linguistic*-nya adalah sama, maka dilakukan pertimbangan dengan mengacu pada elemen-elemen di level yang paling bawah, yaitu jika nilai hasil agregasi pada setiap *linguistic*-nya sama maka dilakukan pertimbangan dengan melihat nilai maksimum jumlah derajat keanggotaan yang nilainya lebih dari 0,5 pada masing-masing *linguistic* yang telah didefinisikan. *Linguistic* yang memiliki jumlah nilai keanggotaan lebih dari 0,5 yang paling besar akan menjadi kategori dari nilai tersebut. Modul-modul tambahan yang telah dibuat untuk membantu dalam memahami makna nilai agregasi *fuzzy* terdiri dari modul_desa, modul_kecamatan, dan

modul_kabupaten. Contoh tampilan kubus data *fuzzy* dapat dilihat pada Gambar 7, untuk kasus penduduk laki-laki dan penduduk

perempuan di kecamatan Dramaga tahun 2006.



Gambar 7 Contoh tampilan kubus data *fuzzy*

KESIMPULAN DAN SARAN

Kesimpulan

Penelitian ini menghasilkan *data warehouse* yang dibangun berdasarkan konsep *fuzzy* serta *data warehouse* untuk data *crisp* sebagai pelengkap penyajian informasi. *Data warehouse* yang dibangun menggunakan konsep *fuzzy* selain melalui tahapan praproses data juga melalui proses *clustering* menggunakan algoritma FCM untuk mendapatkan himpunan *fuzzy* dan nilai derajat keanggotaan tiap atribut. Nilai derajat keanggotaan yang didapatkan merupakan *measure*, sedangkan *linguistic* yang didefinisikan dari jumlah *cluster* merupakan salah satu dari atribut pada level terbawah dalam dimensi yang terkait. *Data warehouse fuzzy* memiliki 4 kubus data yaitu penduduk, rumah tangga, lahan, dan sekolah.

Operasi-operasi yang diimplementasikan adalah operasi *roll-up*, *drill-down*, *slice*, *dice*, dan *pivot*. Operasi untuk *fuzzy* OLAP dilakukan sesuai dengan konsep pengoperasian data multidimensi *fuzzy*. Operator untuk mendapatkan nilai agregasi pada konsep *fuzzy* adalah menggunakan operator *t-norm* atau *t-conorm* yang merupakan implementasi dari operator

minimum atau maksimum. Tampilan data disajikan dalam bentuk *crossstab* dan grafik.

Penelitian ini memberikan informasi berupa data *fuzzy* dan data *crisp*. Data *fuzzy* yang disajikan memberikan informasi natural dengan mendefinisikan variabel *linguistic*. Data *crisp* disajikan sebagai penunjang untuk mendapatkan informasi.

Saran

Saran untuk pengembangan *fuzzy* OLAP adalah mengembangkan sistem yang mengintegrasikan *fuzzy* OLAP dengan beberapa algoritma dalam *data mining* seperti algoritma *association rules*. Sistem tersebut diharapkan akan menyajikan *report* yang lebih lengkap dan lebih bermakna untuk melihat pola atau model data dengan lebih jelas, sehingga lebih informatif untuk mendukung pengambilan keputusan.

DAFTAR PUSTAKA

Alhadj R, Mehmet K. 2003. *Integrating Fuzziness into OLAP for Multidimensional Fuzzy Association Rules Mining*. Proceedings of the Third IEEE International Conference on Data mining.

- Cox, E. 2005. *Fuzzy Modeling and Algorithms for Data mining and Exploration*. USA: Academic Press.
- Han J, Kamber M. 2001. *Data Mining: Concepts and Techniques*. San Diego, USA: Morgan Kaufmann.
- Inmon WH. 1996. *Building the Data warehouse*. New York, USA: John Wiley & Sons.
- Jang JSR, Sun CT, Mizutani Eiji. 1997. *Neuro-Fuzzy and Soft Computing*. London: Prentice-Hall International, Inc.
- Molina Carlos, Ariza LR, S Daniel, Vila M.Amparo. 2006. *A New Fuzzy Multidimensional Model*. IEEE Transaction On Fuzzy System.
- Xie X, Beni G. 1991. *A Validity Measure for Fuzzy Clustering*. IEEE Trans Patt Anal Mach Intel.13(8):841-847.

