

PENERAPAN KONSEP FUZZY DALAM *VARIABLE-CENTERED INTELLIGENT RULE SYSTEM*

(Studi Kasus: Pemilihan Jurusan di Chinese University of Hongkong)

Irfan Subakti dan Oky Wijayanto

Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember (ITS) Surabaya
Email: yifana@gmail.com; oky_wijayanto@yahoo.com

ABSTRAK: *Variable-Centered Intelligent Rule System (VCIRS)* adalah sistem yang terinspirasi dari *Rule-based System (RBS)* dan *Ripple Down Rules (RDR)*. Arsitektur sistem diadaptasi dari RBS dan ia mengambil kelebihan-kelebihan dari RDR. Sistem ini mengorganisasikan basis aturan dalam sebuah struktur yang spesial sehingga kemudahan pembangunan pengetahuan, penelusuran pengetahuan yang kuat, dan perbaikan unjuk kerja sistem yang selalu berkembang dapat diperoleh pada waktu yang sama. Dalam *paper* ini, arsitektur VCIRS dimanfaatkan untuk membangun sebuah sistem pakar yang dapat membantu calon mahasiswa memilih jurusan pada suatu Perguruan Tinggi. Aplikasi sistem pakar ini dapat menangani konsep *fuzzy* seperti *good*, *high*, atau *rather high*, yang merupakan bagian kalimat yang sangat berarti dalam bahasa sehari-hari. Sistem ini dapat menangani nilai yang tepat/teliti, nilai *fuzzy* (atau tidak tepat/tidak teliti), dan jenis pertimbangan gabungan, serta mengizinkan istilah *fuzzy* dan istilah normal untuk digabungkan secara bebas dalam aturan dan fakta. Contoh aplikasi dalam *paper* ini adalah sebuah sistem berbasis aturan yang menggunakan logika *fuzzy* dan bilangan *fuzzy* untuk jenis pertimbangan yang tidak tepat/tidak teliti. Sistem ini menggunakan dua konsep dasar ketidaktepatan/ketidaktelitian, yaitu *fuzziness* dan *uncertainty*. Kasus dalam penelitian ini adalah pemilihan jurusan di Chinese University of Hongkong dalam bentuk basis aturan yang didalamnya mengandung istilah *fuzzy* dan istilah normal. Dari uji coba yang dilakukan, didapat hasil yang sesuai dengan hasil dari sistem Z-II, yaitu alat bantu pembangun sistem pakar komprehensif yang dikembangkan di Chinese University of Hongkong, yang menjadi acuan dari *paper* ini. Sehingga dapat disimpulkan bahwa *fuzzy VCIRS* dapat bekerja dengan baik, serta memberikan hasil yang benar dan dapat dipercaya.

Kata kunci: *rule-based systems, VCIRS, knowledge building, knowledge inferencing, knowledge refining, logika fuzzy, bilangan fuzzy.*

ABSTRACT: *Variable-Centered Intelligent Rule System (VCIRS)* is a system which is inspired by *Rule-based System (RBS)* and *Ripple Down Rules (RDR)*. The system architecture is adapted from RBS, while from RDR this system obtained its advantages. The system organized Rule Base (RB) in a special structure so that easy knowledge building, powerful knowledge inferencing and evolutionally system performance refining can be obtained in the same time. In this paper, the architecture of VCIRS is used to build an expert system for helping students to choose a department at a university. The application of this expert system is able to handle fuzzy concepts (e.g., such as good, high or rather high) which is a prominent part of sentences in natural language. This system is able to cope with exact values, fuzzy (or inexact) values and combined reasoning, allowing fuzzy and normal terms to be freely mixed in the rules and facts. An application example in this paper is a RBS which is employed fuzzy logic and fuzzy number for inexact reasoning. It uses two inexact basic concepts, i.e., fuzziness and uncertainty. A case study presented here is the department admission at Chinese University of Hongkong, formed in a RB containing with fuzzy and normal terms. From experiments performed, there's the proper result obtained comparing with the result from Z-II system (i.e., a comprehensive expert system builder tool developed by Chinese University of Hongkong) which is this paper refers to. So that the conclusion is a fuzzy VCIRS proposed here, is working properly and producing the right and true results.

Keywords: *rule-based systems, VCIRS, knowledge building, knowledge inferencing, knowledge refining, fuzzy logic, fuzzy number.*

PENDAHULUAN

Sistem berbasis pengetahuan yang pengetahuannya disusun oleh aturan-aturan disebut *Rule-based System (RBS)*, disebut juga sebuah sistem pakar (*expert system*). Tempat untuk menyimpan aturan-

aturan disebut sebuah *Knowledge Base (KB)*. Sebuah RBS “tradisional” menggunakan *forward* dan *backward chaining* selama *inferencing* (inferensia penarikan kesimpulan dan penjelasannya). Melalui proses *inference* (penarikan kesimpulan) dapat diperoleh jawaban atas pertanyaan seperti: *What* (apa)

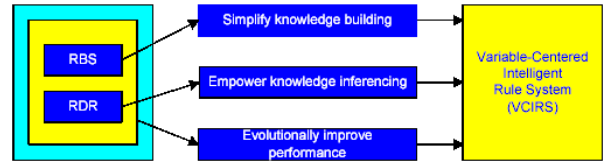
hasil dari proses inferensia? *How* (bagaimana) hal itu dilakukan? *Why* (mengapa) itu bisa dilakukan? Dalam RBS, pakar harus berhubungan dengan pengetahuan dalam sebuah konteks yang spesifik [1].

Ripple Down Rules (RDR) adalah sebuah metode akuisisi pengetahuan yang memaksa interaksi antara pakar dan sebuah *shell* untuk mengakuisisi hanya pengetahuan yang benar [3]. RDR mengatasi masalah utama sistem pakar: pakar tidak lagi harus selalu berhubungan dengan pengetahuan dalam sebuah konteks yang spesifik. RDR mengijinkan akuisisi pengetahuan yang amat sangat cepat dan sederhana tanpa bantuan seorang *knowledge engineer* dengan menyediakan sebuah dukungan yang luas kepada pengguna dalam mendefinisikan aturan-aturan [1]. Hal inilah yang menjadikan sistem RDR memperoleh kekuatan lebih dibandingkan dengan RBS tradisional [2]. Pengguna tidak perlu menguji basis aturan untuk mendefinisikan aturan-aturan baru: pengguna hanya perlu mendefinisikan aturan baru yang mengklasifikasikan dengan benar contoh yang diberikan, dan sistem dapat menentukan dimana aturan baru tersebut seharusnya ditempatkan dalam hirarki *rule*-nya. *Variable-Centered Intelligent Rule System* (VCIRS) terinspirasi dan mengambil keuntungan dari RBS dan RDR [6, 7]. Arsitektur sistem ini mengacu pada arsitektur RBS yang digabungkan dengan keunggulan yang dimiliki oleh RDR.

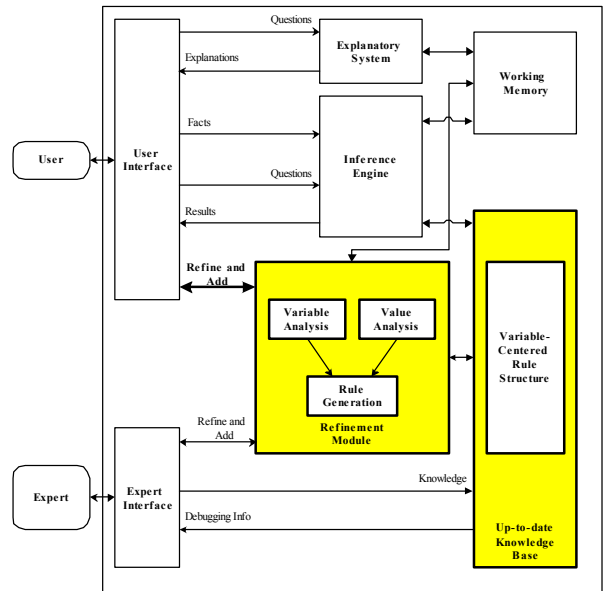
Dalam suatu RBS dapat dijumpai nilai variabel yang mempunyai nilai yang tidak tepat/tidak teliti seperti sangat bagus, agak tinggi, dan sebagainya. Sayangnya, sampai saat ini VCIRS, tidak dilengkapi dengan konsep fuzzy yaitu *fuzzy set*, seperti kategori tinggi/panjang, kondisi baik/bagus, dan sebagainya [7]. Sehingga ada ide konsep *fuzzy set* di atas dimasukkan dalam VCIRS. Selama nilai suatu variabel ada dalam jangkauan nilai fuzzy-nya, maka nilai variabel tersebut dapat diterima.

VARIABLE-CENTERED INTELLIGENT RULE SYSTEM [7]

Metode dari VCIRS dapat digambarkan pada Gambar 1, yang merupakan teknik persilangan dari RBS dan RDR. VCIRS mempunyai struktur yang mengorganisasikan *Rule Base* (RB) sehingga pembangunan pengetahuan yang mudah, inferensia pengetahuan yang berdaya guna dan peningkatan evolusional sistem dapat didapatkan pada waktu yang sama. Sedangkan arsitektur dari VCIRS itu sendiri dapat dilihat pada Gambar 2.



Gambar 1. Metode VCIRS



Gambar 2. Arsitektur Sistem Pakar Menggunakan VCIRS

Pertama, pembangunan pengetahuan dimudahkan dengan langkah-langkah yang sederhana dalam proses pembangunan pengetahuannya. Pengguna tidak perlu mempertimbangkan mengenai struktur KB dan dapat mengupdate KB secara langsung. VCIRS mengijinkan pengguna untuk memperbaiki atau menambahkan *node/rule* ke dalam KB yang telah ada. Seperti dalam RDR, perbaikan *rule* adalah pembuatan sebuah *rule* pengecualian (*exception rule*) untuk membuat benar pengklasifikasian yang salah, dimana penambahan itu mengacu pada penambahan *rule* baru pada level puncak *tree* KB. Sistem memandu pengguna selama proses pembangunan pengetahuan.

Inferensia pengetahuan dipertajam oleh pengetahuan (yaitu, hasil dari analisis variabel dan nilai) dari urutan derajat kepentingan (*important degree*) dan tingkat penggunaan (*usage rate*) dari *case*/kasus pada KB. Mekanisme inferensia RBS dibawa kembali dalam VCIRS, sehingga pengguna mendapatkan lebih banyak jawaban dan penjelasan dari inferensia.

Kinerja sistem ditingkatkan oleh struktur RB yang mendukung analisis variabel (*Variable Analysis*) dan analisis nilai (*Value Analysis*) untuk pembangunan *rule*. Pembangunan *rule* meningkatkan hasil dari inferensia dalam hal cakupan pengetahuan. Lebih

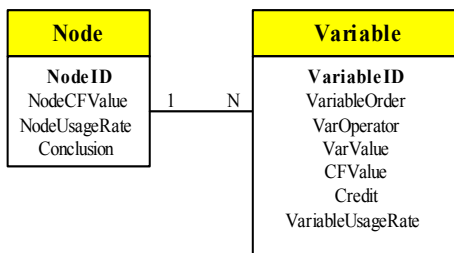
jauh, analisis nilai juga memandu pengguna selama pembangunan dan inferensia pengetahuan. Bersamaan dengan pembangkitan *rule*, kemampuan ini dapat meningkatkan kinerja sistem dalam hal inferensia pengetahuan.

Variable-Centered Rule Structure digunakan untuk merepresentasikan KB dan mendukung *Refinement Module* untuk mengelola KB yang *up-to-date*. Selain itu juga mencatat kasus-kasus dan kejadiannya. Elemen fundamental dari *Variable-Centered Rule Structure* adalah variabel, yang ditempatkan/dipasang oleh pengguna. VCIRS mengelola secara cermat variabel ini mengenai nilainya, struktur dan kejadiannya. Rangkaian dari variabel membentuk *node*, sedangkan rangkaian dari *node* menyusun *rule*. Maka *Variable-Centered Rule Structure* mengandung struktur *rule* dan struktur *node* yang berpusat pada variabel-variabel.

Variable-Centered Rule Structure

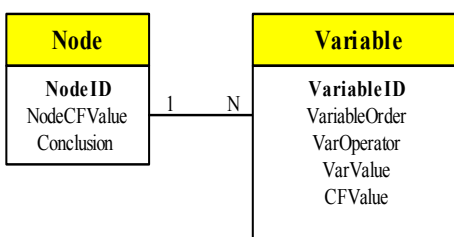
Pada bagian ini akan dijelaskan apa saja yang terlibat dalam *Variable-Centered Rule Structure*.

Node Structure



Gambar 3. Node Structure

Gambar 3 menjelaskan *graf* konseptual dari struktur *node*. Diberikan *case* baru yang disediakan oleh pengguna, dari memori kerja VCIRS memasukkannya ke dalam struktur *node* dan lalu menggunakan struktur *rule* sebagai bagian dari KB yang *up-to-date*. Struktur *node* juga menyimpan kejadian dari variabel-variabel dan *node-node*, untuk *usage assignment* (penugasan tingkat kegunaan). Setiap *case* terdiri dari kumpulan *field* data seperti digambarkan pada Gambar 4.



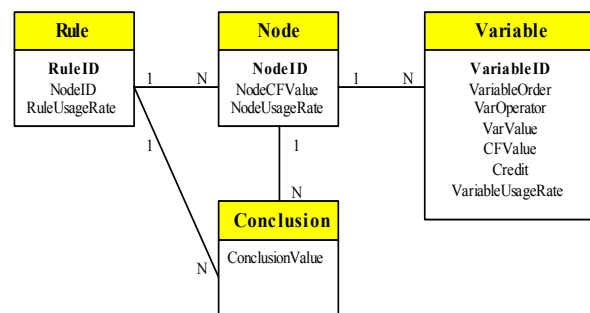
Gambar 4. Case Fields

Setiap kali pengguna memasukkan *case*-nya, struktur *node* mengelola nilai dan posisinya. Informasi ini akan digunakan dalam *usage assignment*.

Dalam VCIRS pengguna dibolehkan untuk memperbaiki atau menambahkan *node* ke dalam KB yang telah ada. Perbaikan *rule* adalah pembuatan dari *rule* pengecualian untuk membenarkan klasifikasi yang salah, sedangkan penambahan *rule* mengacu pada penambahan *rule* baru pada level puncak dari *tree* di KB. Sistem memandu pengguna selama proses pembangunan pengetahuan, akan dijelaskan lebih detil dalam bagian selanjutnya.

Rule Structure

Gambar 5 menggambarkan *graf* konseptual dari *Rule Structure*. Seperti telah dinyatakan sebelumnya, *case* yang dimasukkan oleh pengguna pertama kali disimpan dalam *Node Structure*, lalu digunakan dalam *Rule Structure*.



Gambar 5. Graf Konseptual Rule Structure

Perbaikan Pengetahuan

Ada tiga tugas (*task*) dalam *Refinement Module*: analisis variabel, analisis nilai dan pembangkitan *rule*.

Variable Analysis

Di dalam *Variable-Centered Rule Structure* sistem mengetahui *node* mana yang di-*shared* (sama-sama digunakan) oleh berbagai *rule*, dan variabel mana yang di-*shared* oleh *node*. Semakin banyak *rule* yang memakai suatu *node*, maka *node* tersebut akan semakin penting. Pertimbangan yang sama terjadi pada variabel yang di-*shared* di dalam *node*.

Fakta ini menunjukkan seberapa penting suatu *node*/variabel, sebagai titik awal untuk membangkitkan *rule* baru, bersama dengan analisis nilai.

Value Analysis

Proses analisis nilai, yang disebut dengan *usage assignment* (pemberian nilai kegunaan), adalah untuk menentukan derajat kegunaan dari *rule/node*/variabel

dalam KB. *Usage assignment* menggunakan informasi yang disimpan dalam *Variable-Centered Rule Structure*.

Terdapat tiga *usage degree*. Pertama, *Variable Usage Rate* (VUR) digunakan untuk mengukur tingkat kegunaan dari suatu variabel di dalam *node* yang sedang dan telah digunakan. Kedua, *Node Usage Rate* (NUR) untuk mengukur kegunaan suatu *node* pada pengekseskuan (*firing*) suatu *node*. Yang terakhir adalah *Rule Usage Rate* (RUR) yang mengukur kegunaan suatu *rule* pada pengekseskuan (*firing*) suatu *rule*. Makin besar indikator kegunaan, maka makin bergunalah nilai tersebut dan begitu pula sebaliknya. Baik VUR, NUR maupun RUR adalah bilangan yang bertipe data pecahan (*real*).

Persamaan (1) menghitung VUR untuk variabel ke-*i*, (2) menghasilkan NUR untuk *node* ke-*j*, sedangkan (3) mendefinisikan RUR untuk *rule* ke-*k*.

$$VUR_i = Credit_i \times Weight_i \quad (1)$$

$$NUR_j = \frac{\sum_{i=1}^N VUR_{ij}}{N}, \quad VUR_{ij} \text{ untuk variabel ke-} i \text{ dalam node } j \quad (2)$$

Persamaan (2) dapat dibaca sebagai penjumlahan kumulatif VUR dari sejumlah N variabel dalam *node j* dibagi dengan jumlah N variabel dalam *node j* tersebut. Dimana indeks *i* dalam VUR_{ij} adalah dalam jangkauan nilai 1, ..., N.

$$RUR_k = \frac{\sum_{j=1}^N NUR_{jk}}{N}, \quad NUR_{jk} \text{ untuk node ke-} j \text{ dalam rule } k \quad (3)$$

Persamaan (3) dapat dibaca sebagai penjumlahan kumulatif NUR dari sejumlah N *node* dalam *rule k* dibagi dengan jumlah N *node* dalam *rule k* tersebut. Dimana indeks *j* dalam NUR_{jk} adalah dalam jangkauan nilai 1, ..., N.

Dimana:

- $Credit_i$ = bilangan bertipe data bulat (*integer*) yang merupakan jumlah kejadian dari variabel *i* dalam *Node Structure* (4)
Credit didapatkan dari *Node Structure*. Nilainya akan meningkat saat pengguna membuat *node* yang menyetujui nilai dari *case* lama.
- $Weight_i = NS_i \times CD_i$ (5)
Weight menghitung bobot (*weight*) dari variabel ke *node* yang memilikinya. Tipe datanya adalah pecahan (*real*). Ada dua faktor yang berkontribusi ke bobot dari sebuah variabel. Pertama adalah jumlah *node* yang berbagi (*sharing*) sebuah variabel, dan kedua adalah CD (*Closeness Degree*), yaitu derajat kedekatan sebuah variabel pada sebuah *node*.

- NS_i = bilangan bulat (*integer*) yang merupakan jumlah *node* yang berbagi (*sharing*) variabel *i* (6)

$$CD_i = \frac{VO_i}{TV} \quad (7)$$

CD adalah singkatan dari *Closeness Degree*, yaitu derajat kedekatan sebuah variabel pada sebuah *node*. CD_i dalam *node j*, menghitung derajat kedekatan dari variabel *i* dalam *node j*. Makin dekat sebuah variabel pada konklusi yang dipunyai suatu *node*, makin baiklah variabel tersebut. Sehingga, CD mendasarkan dirinya pada urutan dari variabel dalam suatu *node* (disini, *node* adalah rangkaian dari variabel-variabel). CD adalah bilangan pecahan (*real*) yang dihitung dengan urutan variabel VO, dibagi dengan total variabel TV, yang dimiliki oleh sebuah *node*.

- VO_i = urutan dari variabel *i* dalam suatu *node* (8)
- TV = total variabel yang dimiliki oleh suatu *node* (9)

Rule Generation

Pembangkitan *rule* bekerja berdasarkan hasil dari analisis variabel dan nilai. Perlu dicatat bahwa dari analisis variabel dihitung *important degree* dari suatu *node*/variabel, sedangkan dari analisis nilai didapatkan *usage degree* dari *rule*/node/variabel.

Informasi mengenai *shared node*/variabel dari analisis variabel berguna untuk memilih kandidat yang baik untuk membuat kombinasi. *Shared node*/variabel yang paling tinggi nilainya berarti bahwa ialah yang merupakan *node*/variabel terpenting dalam KB yang ada, sebab digunakan di banyak tempat pada struktur saat ini. *Usage degree* dengan nilai tertinggi yang diperoleh dari analisis nilai mengandung arti bahwa *node*/variabel tersebut memiliki kejadian (*occurrence*) tertinggi di dalam struktur.

Kombinasi variabel mengkombinasikan variabel untuk menghasilkan *node* baru, sedangkan kombinasi *node* mengkombinasikan *node* untuk menghasilkan *rule* baru. Kombinasi ini dapat dilakukan asalkan urutan dari variabel/*node* yang akan dihasilkan tidak menyalahi urutan variabel/*node* yang telah ada dalam KB. *Rule* dibangkitkan dengan mengkombinasikan *node-node* berdasarkan urutan relatif dari *node-node* yang telah ada dalam *rule*.

Selama proses pembangkitan *rule* juga dapat dilakukan pembangkitan *node*. Variabel terakhir dari kandidat *node* didapatkan dari variabel yang terpenting (*most important variable*). *Node* dibangkitkan dengan mengkombinasikan variabel-variabel menurut urutan relatif variabel yang telah ada dalam *node*.

Pembangunan Pengetahuan

Gambar 6 menjelaskan algoritma dari pembangunan pengetahuan.

I. Jika *node* yang layak ditemukan sistem akan memberikan pengguna satu dari beberapa pilihan berikut ini.

1. Jika ia tidak setuju dengan *node* saat itu
2. Ia dapat membuat *node* baru pada level puncak KB Secara otomatis
3. Sistem akan membuatkan *rule* baru untuk *node* baru ini. Variabel-variabel yang tak teretujui dari *case* yang dimasukkan pengguna akan disimpan di bawah posisi *node* baru.
4. Jika ia setuju dengan satu atau lebih variabel dalam *node* saat itu.
5. Tetapi tidak menyetujui dengan yang lainnya; ia dapat membuat *node* pengecualian yang mengandung variabel-variabel yang tidak disetujuinya tadi. Variabel yang teretujui dari *case* yang dimasukkan pengguna akan disimpan dalam kolom *Credit* dari *Node Structure* di bawah posisi dari *node* lama (*parent node*).
6. Sedangkan variabel yang tak teretujui akan disimpan di bawah posisi *node* baru.
7. Ia dapat memilih untuk melanjutkan penelusuran KB tanpa merubah apa pun. Disini *Variable-Centered Rule Structure* tak mendapat apa pun dari pengguna. Kerja sistem jika seperti ini serupa dengan sebuah proses inferensia dan ia membolehkan pengguna untuk menguji KB.
8. Yaitu mekanisme verifikasi sambil jalan (*verification-on-the-fly*).

II. Jika tak ada satu pun *node* yang ditemukan sistem akan meminta pengguna untuk membuat *node* baru pada level puncak dari KB dan *case* yang dimasukkan pengguna akan disimpan di bawah posisi *node* baru tadi.

Gambar 6. Algoritma Pembangunan Pengetahuan

Konsep Fuzzy

Pada bagian ini akan dijelaskan hal-hal yang berhubungan dengan konsep *fuzzy*.

Bilangan Fuzzy [5]

Ketidakpastian *fuzzy* (*fuzzy uncertainty*) dimodelkan dengan bilangan *fuzzy* yang merepresentasikan konsep *around 0.8*, *close to 1.0*, dan lain-lain.

Formula untuk beberapa operator aritmatika pada bilangan *fuzzy* adalah sebagai berikut:

- Penjumlahan bilangan *fuzzy* +:

$$\mu_{A+B}(z) = \bigvee_{z=x+y} (\mu_A(x) \wedge \mu_B(y))$$

- Pengurangan bilangan *fuzzy* -:

$$\mu_{A-B}(z) = \bigvee_{z=x-y} (\mu_A(x) \wedge \mu_B(y))$$

- Perkalian bilangan *fuzzy* *:

$$\mu_{A*B}(z) = \bigvee_{z=x*y} (\mu_A(x) \wedge \mu_B(y))$$

- Pembagian bilangan *fuzzy* /:

$$\mu_{A/B}(z) = \bigvee_{z=x/y} (\mu_A(x) \wedge \mu_B(y))$$

- Minimum dari dua buah bilangan *fuzzy* *min_fn*:

$$\mu_{\min_fn(A,B)}(z) = \bigvee_{z=x \wedge y} (\mu_A(x) \wedge \mu_B(y))$$

- Maksimum dari dua buah bilangan *fuzzy* *max_fn*:

$$\mu_{\max_fn(A,B)}(z) = \bigvee_{z=x \vee y} (\mu_A(x) \vee \mu_B(y))$$

di mana:

A dan B: bilangan *fuzzy*

μ_A adalah fungsi keanggotaan dari *fuzzy subset* A, $\mu_A: \rightarrow [0, 1]$.

x, y, dan z: bilangan *real*

\bigvee dan \vee : diambil yang maksimum

\wedge : ambil yang minimum

Logika Fuzzy

Fuzzy Composition

Aturan (*rule*): *If the price is high, then the profit is good*, dengan konsep *fuzzy high* dan *good*, dapat dimodelkan dengan sebuah relasi *fuzzy* R. Misalkan A_1 dan A_2 adalah *fuzzy set*, masing-masing merepresentasikan konsep *high* dan *good* terhadap *price* dan *profit*. Didapatkan relasi *fuzzy* R, yang direpresentasikan dengan sebuah matrix, dengan melakukan operasi *fuzzy* pada A_1 dan A_2 , dan dinyatakan sebagai *vector*. Banyak peneliti telah mengusulkan metode komputasi relasi *fuzzy* R.

Jika sebuah fakta *The price is very high* sesuai dengan *rule* di atas, konsep *fuzzy very high* dapat ditunjukkan dengan sebuah *fuzzy set* F, yang diperoleh dengan melakukan sebuah operasi aritmatika (sebuah operasi kuadrat dalam kasus ini, yang menggunakan *truth-quantified proposition*) pada A_1 . Kemudian dihitung *fuzzy set* C, merepresentasikan istilah *fuzzy* dalam konklusi, dengan menggunakan sebuah operator *fuzzy* yang disebut *composition* (komposisi penyusunan) (dilambangkan dengan o) pada F dan R.

Formula untuk *fuzzy composition* adalah sebagai berikut:

$$C = F \circ R$$

$$\mu_C(x) = \bigvee_w (\mu_F(w) \wedge \mu_R(w, x))$$

dimana $\mu(x)$ adalah sebuah fungsi keanggotaan (*membership function*) dan w serta x adalah elemen-elemen dalam semesta pembicaraan (*universe of discourse*).

Sebagai hasilnya, *fuzzy set* C akan mengindikasikan *very good* dan konklusi *The profit is very good* akan digambarkan. Operasi ini merupakan sebuah inferensia sederhana dari sebuah fakta dan sebuah *rule* yang keduanya berisi istilah *fuzzy*.

Evaluasi Rule [4]

Misalnya terdapat sebuah *rule* dan fakta:

rule: IF A is V_1 THEN C is V_2 (FN₁)

fact: A is V_1' (FN₂)

conclusion: C is V_2' (FN₃)

dimana:

A : objek *antecedent*

C : objek konklusi (*consequent*)

FN₁ : bilangan *fuzzy* yang menunjukkan *uncertainty* dari *rule*

FN₂ : bilangan *fuzzy* yang menunjukkan *uncertainty* dari fakta

FN₃ : bilangan *fuzzy* yang menunjukkan *uncertainty* dari konklusi

V_1, V_2, V_1', V_2' : *value/nilai*

Jenis objek yang mungkin beserta evaluasinya adalah sebagai berikut:

- Jika objek A dalam *antecedent* adalah bukan/*non fuzzy*, maka V_1 dan V_1' harus sama secara *atomic symbol* untuk menggunakan *rule* ini. Karena itu, V_2' dalam konklusi sama dengan V_2 , dan *fuzzy uncertainty* FN₃ dari *conclusion* dihitung dengan perkalian bilangan *fuzzy* dari FN₁ dan FN₂: FN₃ = FN₁ * FN₂
- Jika A dan C adalah objek *fuzzy*, maka V_1 dan V_2 direpresentasikan dengan *fuzzy set* F_1 dan F_2 . Sebuah relasi *fuzzy* R dapat dibentuk dengan melakukan beberapa operasi *fuzzy* pada F_1 dan F_2 . Metode *default* yang diangkat dalam sistem Z-II untuk membentuk relasi *fuzzy* R adalah pendekatan R_{SG} yang diajukan oleh Mizumoto, Fukami, dan Tanaka, yang lebih mendekati intuisi manusia dan pertimbangan daripada metode lain. R_{SG} = $(F_1 \times V \rightarrow_S U \times F_2) \wedge (\sim F_1 \times V \rightarrow_G U \times \sim F_2)$ dimana:

$$\mu_{F_1}(u) \rightarrow_S \mu_{F_2}(v) = \begin{cases} 1 & \text{if } \mu_{F_1}(u) \leq \mu_{F_2}(v) \\ 0 & \text{if } \mu_{F_1}(u) > \mu_{F_2}(v) \end{cases}$$

$$\mu_{F_1}(u) \rightarrow_G \mu_{F_2}(v) = \begin{cases} 1 & \text{if } \mu_{F_1}(u) \leq \mu_{F_2}(v) \\ \mu_{F_2}(v) & \text{if } \mu_{F_1}(u) > \mu_{F_2}(v) \end{cases}$$

U dan V: *universe of discourse* dari F_1 dan F_2

u dan v: elemen dari sebuah *fuzzy set*

\wedge : irisan (*intersection*) dari dua buah relasi *fuzzy*

\sim : komplemen sebuah *fuzzy set*

μ : fungsi keanggotaan (*membership function*)

x: perkalian kartesian dua buah *fuzzy set*

V_1' dalam fakta seharusnya juga berupa nilai *fuzzy* yang direpresentasikan dengan sebuah *fuzzy set* F_1' . *Fuzzy set* F_2' dari V_2' dalam *conclusion* diperoleh dengan melakukan sebuah operasi *fuzzy composition* (dilambangkan dengan \circ) pada F_1' dan R: $F_2' = F_1' \circ R$. Perhitungan *fuzzy uncertainty* FN₃ pada *conclusion* adalah sama seperti pada poin 1.

- Jika A adalah *fuzzy* dan C adalah *nonfuzzy*, maka V_2' dalam *conclusion* harus sama dengan V_2 . Tetapi, *fuzzy uncertainty* FN₃ diperoleh dengan perkalian bilangan *fuzzy* dari FN₁, FN₂, dan persamaan (*similarity*) M antara FN₁ dan FN_{1'}, yaitu *fuzzy set* dari V_1 dan V_1' .

$$FN_3 = (FN_1 * FN_2 * M)$$

Similarity M dihitung dengan algoritma di bawah ini:

$$\text{IF } N(F_1|F_1') > 0.5$$

$$\text{THEN } M = P(F_1|F_1')$$

$$\text{ELSE } M = (N(F_1|F_1') + 0.5) * P(F_1|F_1')$$

Di bawah ini adalah formula pengukuran *possibility* dan *necessity* antara data *fuzzy* dan *fuzzy pattern*:

$$\text{possibility: } P(F_1|F_1') = \max(\min(\mu_{F_1}(w), \mu_{F_1'}(w)))$$

$$\text{necessity: } N(F_1|F_1') = 1 - P(\sim F_1|F_1')$$

dimana μ adalah *membership distribution function*, w adalah elemen dalam *universe of discourse* dari *fuzzy set*, dan $\sim F_1$ adalah komplemen dari F_1 .

Dalam keadaan normal *necessity* mempunyai hubungan antara dua buah *fuzzy set*:

$$N(F_1|F_1') > 0.5 \Leftrightarrow F_1' \text{ adalah sebuah } \textit{concentration} \text{ dari } F_1.$$

$$N(F_1|F_1') = 0.5 \Leftrightarrow F_1' \text{ adalah sebuah } \textit{duplicate} \text{ dari } F_1.$$

$$N(F_1|F_1') < 0.5 \Leftrightarrow F_1' \text{ adalah sebuah } \textit{dilation} \text{ dari } F_1.$$

- Rule* dengan *consequent propositions* majemuk. Dalam sistem pakar, bagian *consequent* dari sebuah *rule* dapat terdiri dari banyak preposisi (C_1, C_2, \dots, C_n) dengan konjungsi AND diantaranya. Mereka dapat diperlakukan sebagai banyak *rule* dengan satu *consequent*. Sehingga bentuk *rule* di bawah ini:

IF antecedent-propositions, THEN C_1 AND C_2
AND ... C_n

adalah ekuivalen dengan *rule-rule* di bawah ini:

IF antecedent-propositions, THEN C_1

IF antecedent-propositions, THEN C_2

...

IF antecedent-propositions, THEN C_n

5. *Rule* dengan antecedent propositions majemuk

rule: IF A_1 AND/OR A_2 , THEN C is V_3 (FN_R)

fact: A_1' (FN_1)

A_2' (FN_2)

conclusion: C is V_3' (FN_C)

algoritma:

fuzzy set yang merepresentasikan V_3' dalam *conclusion* C diperoleh dengan melakukan *fuzzy union* (untuk operator AND) atau *fuzzy intersection* (untuk operator OR) dari *fuzzy set* F_1 dan F_2

dimana:

fuzzy set F_1 diperoleh dari *composition operation* pada *rule* tunggal (IF A_1 , THEN C is V_3) dan fakta A_1' , sedangkan *fuzzy set* F_2 diperoleh dari *composition operation* pada *rule* tunggal (IF A_2 , THEN C is V_3) dan fakta A_2' .

A_1, A_2 : antecedent proposition yang dapat berupa tunggal atau majemuk

C : objek dalam consequent proposition

A_1', A_2' : preposisi data (fakta)

V_3, V_3' : nilai

Fuzzy intersection D dari F_1 dan F_2 diperoleh dengan mencari nilai keanggotaan minimum dari F_1 dan F_2 , atau dapat dituliskan:

IF D adalah *fuzzy intersection* pada F_1 dan F_2 ,

THEN $\mu_D(x) = \min(\mu_{F_1}(x), \mu_{F_2}(x))$

Fuzzy union D dari F_1 dan F_2 diperoleh dengan mencari nilai keanggotaan maksimum dari F_1 dan F_2 , atau dapat dituliskan:

IF D adalah *fuzzy union* pada F_1 dan F_2 ,

THEN $\mu_D(x) = \max(\mu_{F_1}(x), \mu_{F_2}(x))$

D, F_1, F_2 : *fuzzy set*

Fuzzy uncertainty conclusion disimpulkan dari *rule* dengan antecedent proposition majemuk dihitung dengan operator aritmetika bilangan *fuzzy* dalam formula:

$FN_C = (\min(FN_1, FN_2)) * FN_R$

FN_R : *fuzzy uncertainty* dari *rule*

FN_1, FN_2 : *fuzzy uncertainty* dari fakta

FN_C : *fuzzy uncertainty* dari *conclusion*

6. *Evidence combination*

Dalam beberapa kasus, dua atau lebih *rule* mempunyai consequent proposition yang sama.

Sebagai contoh:

rule: r_1 – if A_1 then C

r_2 – if A_2 then C

fact: $-A_1', A_2'$

conclusion: C_R , diperoleh dari:

$-C'$ (FN_1)

dan $-C''$ (FN_2)

dimana:

r_1, r_2 : kode *rule*

A_1, A_2 : antecedent proposition

C : consequent proposition

C', C'' : *conclusion* dari r_1 & A_1' dan r_2 & A_2'

FN_1, FN_2 : *fuzzy uncertainties* dari *conclusion*

Jika objek yang terlibat dalam consequent proposition adalah *fuzzy*, *fuzzy set* yang bersesuaian dengan *conclusion* C' dan C'' , diperoleh dari *fuzzy intersection* antara keduanya untuk memperoleh *fuzzy set* yang bersesuaian dengan konklusi gabungan C_R .

Fuzzy uncertainties dari *conclusion* C', C'' juga bersama-sama membentuk *uncertainty* keseluruhan FN_R untuk C_R .

$FN_R = FN_1 + FN_2 - (FN_1 * FN_2)$

dimana

FN_R : *fuzzy uncertainty* gabungan

FN_1, FN_2 : *fuzzy uncertainties* dari C' dan C''

EVALUASI HASIL PROSES INFERENSIA

Sistem Z-II mempunyai sekitar 60 *rule* dalam KB-nya yang didesain untuk membantu calon mahasiswa memilih jurusan di Chinese University of Hong Kong (sebagian dari *rule-rule* ini ada di Gambar 8) [4].

Dalam uji coba proses inferensia pada sistem Fuzzy VCIRS dimasukkan contoh fakta sebagai berikut:

1. *The fact that you have a risk-taking personality is false.* (1.0)
2. *Your interest in the business environment is very low.* (about 0.9)
3. *Your Chinese result is fair.* (1.0)
4. *The mark you obtained in the English Usage Test is 76.0.* (1.0)
5. *Your mathematics result is good.* (1.0)
6. *The preferred career as a teacher is true.* (1.0)
7. *The preferred career as a researcher is true.* (1.0)
8. *Your interest in analyzing human behavior is medium.* (around 0.8)
9. *Your interest in reading fiction is more or less low.* (1.0)
10. *The preferred carrer as a doctor is false.* (1.0)
11. *Your interest in analyzing the human body is low.* (0.7)
12. *Your chemistry result is fair.* (1.0)

13. *Your biology result is fair.* (1.0)
14. *Your physics result is very good.* (1.0)
15. *Your interest in manipulating mathematical symbol is rather high.* (1.0)
16. *The preferred career as an electronics engineer is false.* (1.0)
17. *Your interest in building electronic kits is medium.* (1.0)
18. *The preferred career as a programmer is true.* (1.0)
19. *Your interest in writing computer programs is high.* (1.0)
20. *Your interest in computers is high.* (1.0)
21. *Your interest in reading science books or magazines is very high.* (1.0)
22. *Your interest in performing chemical experiments is medium.* (1.0)
23. *Your interest in analyzing chemical substances is more or less high.*
24. *Your interest in observing animals is high.* (1.0)
25. *Your interest in observing plants is high.* (1.0)

Dengan menggunakan fakta di atas dalam uji coba pada sistem Fuzzy VCIRS yang dibuat dalam *paper* ini, akhirnya diperoleh hasil proses inferensia seperti tampak pada Gambar 7.

It is 0.5048625 certain that-the department of medicine
It is 0.71455 certain that - the department of arts
It is 0.516825 certain that-the department of business
It is 0.70366 certain that - the department of social science
It is 0.9516640000000001 certain that - the department of computer science
It is 0.9723999999999999 certain that - the department of physics
It is 0.8005503999999999 certain that - the department of biology
It is 0.6478473297696313 certain that - the department of chemistry
It is 0.8193877190723977 certain that - the department of mathematics
It is 0.9135000000000001 certain that - the department of electronics

Gambar 7. Hasil Proses Inferensia pada Fuzzy VCIRS

Dari hasil proses inferensia di atas dapat diketahui bahwa berdasarkan fakta yang dimasukkan diperoleh kesimpulan bahwa calon mahasiswa tersebut sebaik-

nya memilih jurusan Fisika dengan tingkat keyakinan paling tinggi yaitu 0.97.

Setelah membandingkan hasil konklusi di atas dengan hasil yang diperoleh dengan menggunakan sistem Z-II, ternyata terdapat selisih nilai *certainty* yang cukup signifikan pada konklusi yang mempunyai nilai: *the department of biology, the department of chemistry, dan the department of mathematics.*

Variable-Centered Rule Structure belum dilengkapi kemampuan untuk menangani operator logika selain AND, yaitu operator logika OR. Sehingga pada saat proses transformasi *node structure* ke dalam *RBS Structure*, menggunakan operator logika AND pada *rule* yang mempunyai *antecedent proposition* majemuk, termasuk *rule* yang semestinya menggunakan operator OR pada salah satu *antecedent proposition*-nya. Padahal penghitungan *fuzzy set* dan *fuzzy uncertainty* konklusi menggunakan formula yang berbeda berdasarkan operator logika yang digunakan pada *antecedent proposition* dalam sebuah *rule*. Nilai *fuzzy uncertainty* yang dihitung jika menggunakan logika OR adalah lebih besar daripada nilai *fuzzy uncertainty* yang dihitung jika menggunakan logika AND pada *antecedent proposition* dalam sebuah *rule* (karena jika *antecedent proposition* menggunakan logika OR, nilai *fuzzy uncertainty* konklusi dihitung dengan formula bilangan *fuzzy* maksimum (max_{fn})).

Perbedaan yang juga signifikan terdapat pada konklusi yang mempunyai nilai *the department of mathematics*. Hal ini disebabkan karena dalam *paper* ini digunakan asumsi bahwa *fuzzy set fair or good* merupakan *fuzzy set* dasar (bukan merupakan *modifier* atau dua *fuzzy set* yang dihubungkan dengan operator logika OR). Sehingga jika menggunakan definisi *fuzzy set* yang berbeda dapat mempengaruhi nilai *necessity* yang merupakan indikasi apakah *fuzzy set* fakta merupakan *concentration, duplicate, atau dilation* dari *fuzzy set* objek *antecedent*.

Secara garis besar proses inferensia di atas telah sesuai dengan formula yang digunakan dalam sistem Z-II, sehingga dapat dipastikan bahwa sistem yang dibangun dalam *paper* ini dapat bekerja dengan baik dan memberikan hasil yang benar.

Gambar 8 menampilkan sebagian dari *rule-rule* yang digunakan oleh sistem Z-II dalam membantu calon mahasiswa memilih jurusan di Chinese University of Hong Kong.

RULE CODE: exam-r1
 IF your biology result is very good AND
 your chemistry result is good AND
 (your mathematics result is fair or good OR
 your physics result is fair or good)
 THEN your overall examination performance in
 biology should be good
 WITH CERTAINTY → 1.0

RULE CODE: exam-r2
 IF your chemistry result is very good AND
 (your mathematics result is fair or good OR
 your physics result is fair or good)
 THEN your overall examination performance in
 chemistry should be good
 WITH CERTAINTY → 1.0

RULE CODE: exam-r3
 IF your physics result is very good AND
 your mathematics result is good
 THEN your overall examination performance in
 physics should be good
 WITH CERTAINTY → 1.0

RULE CODE: exam-r4
 IF your mathematics result is very good AND
 your physics result is good
 THEN your overall examination performance in
 computer science should be good
 WITH CERTAINTY → 1.0

RULE CODE: test-r2
 IF the mark you obtained in the English
 Usage Test is < 70.0 AND
 the mark you obtained in the English
 Usage Test is ≥ 50.0
 THEN your English result should be fair
 WITH CERTAINTY → 1.0

RULE CODE: test-r3
 IF the mark you obtained in the English
 Usage Test is < 80.0 AND
 the mark you obtained in the English
 Usage Test is ≥ 70.0
 THEN your English result should be quite good
 WITH CERTAINTY → 1.0

RULE CODE: test-r4
 IF the mark you obtained in the English
 Usage Test is < 90.0 AND
 the mark you obtained in the English
 Usage Test is ≥ 80.0
 THEN your English result should be good
 WITH CERTAINTY → 1.0

RULE CODE: test-r5
 IF the mark you obtained in the English
 Usage Test is ≥ 90.0
 THEN your English result should be very good
 WITH CERTAINTY → 1.0

KESIMPULAN

Dari hasil uji coba yang telah dilakukan dapat diambil kesimpulan:

1. *Variable-Centered Rule Structure* dapat menerima objek *fuzzy* selain objek normal sehingga dapat menangani nilai yang tidak tepat/tidak teliti.
2. *Variable-Centered Rule Structure* masih belum dapat menangani operator logika selain AND, misalnya operator logika OR.
3. Pembangkitan *node* atau *rule* masih dilakukan secara manual yaitu pengguna memilih variabel atau *node* yang merepresentasikan kesalahan semantik.
4. Bilangan *fuzzy* memakai *uncertainty* yang digunakan untuk menyatakan persentase keyakinan seorang calon mahasiswa cocok masuk ke suatu jurusan berdasarkan hasil ujian calon mahasiswa, kepantasan pada bidang pekerjaan di masa depan, dan ketertarikan calon mahasiswa pada suatu jurusan.

DAFTAR PUSTAKA

1. Compton, P., Edwards, G., Kang, B., Lazarus, L., Malor, R., Menzies, T., Preston, P., Srinivasan, A. and Sammut, S., "Ripple down rules: possibilities and limitations," in Boose, J.H. and Gaines, B.R., (Eds.), *Proceedings of the Sixth AAI Knowledge Acquisition for Knowledge-Based Systems Workshop*, Calgary, Canada, University of Calgary, 1991, pp. 6-1-6-20.
2. Ho, V., Wobcke, W. and Compton, P., "EMMA: an e-mail management assistant," in Liu, J., Faltings, B., Zhong, N., Lu, R., Nishida, T. (Eds.), in *Proc. of IEEE/WIC International Conference on Intelligent Agent Technology*, Los Alamitos, CA, 2003, pp. 67-74.
3. Kang, B., Compton, P. and Preston, P., "Multiple classification ripple down rules: evaluation and possibilities," in Gaines, B. and Musen, M., *Proceedings of the 9th AAI-Sponsored Banff Knowledge Acquisition for Knowledge-Based Systems Workshop*, Banff, Canada, University of Calgary, 1995, pp. 17.1-17.20.
4. Leung, K.S. and Lam, W., "Fuzzy Concepts in Expert Systems", *IEEE Computer Society*, 1989.
5. Ross, T.J., *Fuzzy Logic with Engineering Application*, McGraw-Hill, International Editions, USA, 1997.
6. Subakti, I., "A variable-centered intelligent rule system," *Master Thesis*, Department of Computer Science and Information Engineering, College of Electrical and Computer Engineering, National Taiwan University of Science and Technology, Taiwan, 2005.

Gambar 8. Sebagian Rule-Rule dalam Sistem Z-II

7. Subakti, I., "A variable-centered intelligent rule system," *Proc. of the 1st Annual International Conference: Information and Communication Technology Seminar 2005 (ICTS2005)*, Vol. 1, No. 1, August 2005, Institute Technology of Sepuluh Nopember (ITS), Surabaya, Indonesia, 2005, pp. 167-174.