

Penerapan Fuzzy Linguistic Multi-Expert Pada Penentuan Nilai Ketidakpastian Sistem Pakar Berbasis Rule Based Reasoning (RBR)

Rusdi Efendi

Jurusan Teknik Informatika Fasilkom Unsri

Email : Rusdiefendi8@gmail.com

Abstract

Expert system is a branch of artificial intelligence (AI), which use specialized knowledge to solve a problem at the level of experts. expert is someone who has keahliaan in a particular discipline. In the expert system to make a decision rule is often found answers that do not have full certainty. This uncertainty may be a probability or a probability that depends on the outcome of an event. The probability / uncertainty in expert systems caused by the inability of experts to formulate a rule for sure. Usually its value is determined by the experts themselves. Sometimes an expert is very difficult to determine an estimate of the value of a symptom of uncertainty and it will be easier to determine if the use of the term "often, sometimes, rarely. The method used in the study "the application of fuzzy linguistic multi-expert on determining the value of uncertainty based expert system rule-based reasoning (RBR)" This will output a value of uncertainty of expert system-based rules with input data that is processed in the form of symptom data / evidence that using words fuzzy.

Keywords- *expert system, the value of the uncertainty, fuzzy linguistic, rule-based reasoning*

Abstrak

Sistem pakar merupakan cabang dari artificial intelligence (AI) yang menggunakan pengetahuan khusus untuk memecahkan suatu permasalahan pada tingkat pakar. pakar adalah seseorang yang memiliki keahlian pada suatu disiplin ilmu tertentu. Dalam sistem pakar untuk membuat suatu aturan keputusan sering ditemukan jawaban yang tidak memiliki kepastian penuh. Ketidakpastian ini bisa berupa probabilitas atau kebolehjadian yang tergantung dari hasil suatu kejadian. Probabilitas/ketidakpastian dalam sistem pakar disebabkan oleh ketidakmampuan seorang pakar untuk merumuskan suatu aturan secara pasti. Biasanya nilainya ditentukan oleh pakar itu sendiri. Kadang-kadang seorang pakar sangat susah untuk menentukan perkiraan dari nilai ketidakpastian suatu gejala dan akan lebih mudah untuk menentukannya jika menggunakan istilah "sering, kadang-kadang, jarang. Metode yang digunakan dalam penelitian "penerapan fuzzy linguistic multi-expert pada penentuan nilai ketidakpastian sistem pakar berbasis rule based reasoning (rbr)" ini akan memberikan output berupa nilai ketidakpastian dari sistem pakar yang berbasis aturan dengan inputan data yang diolah berupa data gejala/ evidence yang menggunakan kata-kata fuzzy.

Kata Kunci— *sistem pakar, nilai ketidakpastian, fuzzy linguistic, rule based reasoning*

1. PENDAHULUAN

Sebagaimana kita tahu bahwa sistem informasi memegang peranan yang sangat penting dalam kehidupan kita. Sebagai salah satu contohnya, kita dapat melihat manager dari perusahaan-perusahaan yang ada dapat memperoleh sejumlah informasi yang sangat penting dengan adanya sistem informasi. Pada dasarnya, sistem informasi terbagi-bagi menjadi beberapa bagian misalnya Sistem Informasi Akuntansi, Sistem Informasi Manajemen, dan sebagainya. Secara garis besarnya sistem yang ada disesuaikan untuk kebutuhan informasi dari sejumlah besar manager. Dalam banyak kasus informasi ini kurang memadai untuk membuat keputusan yang spesifik untuk memecahkan permasalahan yang spesifik. Oleh karena itulah Sistem Pendukung Keputusan dibuat sebagai suatu cara untuk memenuhi kebutuhan ini [1].

Apabila data-data atau informasi yang diberikan, baik oleh pengambil keputusan tidak dapat disajikan dengan lengkap, mengandung ketidakpastian atau ketidak konsistenan maka akan menyebabkan terjadi kesalahan dalam proses pengambilan keputusan. Bahkan yang lebih parah lagi, seandainya proses pengambilan keputusan tersebut di lakukan dalam sebuah kelompok besar yang otomatis melibatkan banyak orang. Dalam keadaan seperti itu akan terlihat dengan jelas adanya perbedaan bahwa setiap individu akan dicirikan oleh persepsi, sikap dan motivasinya sendiri, siapa yang mau menerima dan mengakui suatu permasalahan secara umum dalam usaha untuk mendapatkan suatu keputusan yang bersifat kolektif [2].

Komputer yang pada awalnya hanya digunakan untuk melakukan pengolahan data, saat ini telah berkembang fungsinya untuk membantu dalam pengambilan keputusan dengan tujuan agar komputer memiliki kemampuan seperti yang dimiliki manusia. Ilmu pengetahuan yang mempelajari bagaimana membuat komputer dapat bertindak dan memiliki kecerdasan seperti manusia disebut kecerdasan buatan. Salah satu bidang dari kecerdasan buatan adalah sistem pakar .

Sistem pakar adalah cabang dari *Artificial Intellegence* yang membuat ekstensi khusus untuk spesialisasi pengetahuan guna memecahkan suatu permasalahan tertentu pada level *humam expert*. *Humam Expert* adalah seseorang yang ahli dalam suatu bidang ilmu pengetahuan tertentu, ini berarti bahwa *humam expert* memiliki suatu pengetahuan atau skill khusus yang tidak dimiliki oleh orang lain. Dalam system pakar untuk membuat suatu aturan keputusan sering ditemukan jawaban yang tidak memiliki kepastian penuh. Ketidakpastian ini bisa berupa probabilitas atau kebolehjadian yang tergantung dari hasil suatu kejadian. Hasil yang tidak pasti disebabkan oleh dua factor yaitu aturan yang tidak pasti dan jawaban pengguna yang tidak pasti atas suatu pertanyaan yang diajukan oleh system[3].

Probabilitas/ketidakpastian dalam sistem pakar disebabkan ketidakmampuan seorang pakar untuk merumuskan suatu aturan secara pasti. Misalnya jika seseorang mengalami sakit kepala, demam dan bersin-bersin ada kemungkinan orang tersebut terserang penyakit flu, tetapi bukan berarti apabila seseorang mengalail gejala tersebut pasti terserang penyakit flu. Untuk mengatasi hal tersebut dalam sistem pakar maka digunakan beberapa algoritma untuk menghitung nilai ketidakpastian.[4].

Sebagai contoh : jika pasien mengalami demam dan bintik-bintik maka pasien menderita cacar dengan nilai ketidakpastian : 80 %, jika dibuat rule dalam sistem pakar, maka bentuk Rule Based nya menjadi :

”IF pasien demam and bintik-bintik then cacar dengan Keyakinan : 0.8”.

Tetapi terkadang untuk mendapatkan nilai ketidakpastian dalam rule diatas (80%) terkadang pemahaman antara pakar dengan kita tidaksama. Sehingga dari pengalaman yang terjadi nilai ketidakpastian yang diinputkan kedalam rule pada sistem tersebut bukanlah nilai asli dari pakar tetapi banyak yang merupakan hasil rekayasa.

Untuk mengatasi hal tersebut maka penulis berinisiatif untuk menerapkan metode *Fuzzy Multi Expert Multi Attribute Decision Making*. Didalam metode ini nantinya setiap

rule tidak disertakan dengan nilai ketidakpastian seperti pada contoh rule diatas, tetapi cukup mengisi tingkat intensitas dari masing-masing gejala dari penyakit tersebut dengan menggunakan kata-kata yang bersifat fuzzy.

Metode yang biasa digunakan dalam sistem pakar biasanya hanya untuk menentukan nilai ketidakpastian dari seorang pakar sedangkan untuk menentukan nilai ketidakpercayaan dari banyak pakar kita dapat menggunakan metode gabungan antara sistem pendukung keputusan dengan teori yang dimiliki oleh sistem pakar tersebut. Salah satu alternatif metode yang digunakan adalah *Fuzzy Multi Expert MADM* dikarenakan biasanya diagnosa dari pakar lebih cenderung menggunakan kata-kata yang bersifat fuzzy, misalnya : untuk menentukan frekuensi dari gejala yang muncul sering menggunakan kata : sering, kadang-kadang, tidak pernah. Salah satu sistem pakar yang pernah dibuat dengan menggunakan penalaran teori Fuzzy MADM untuk melakukan tindakan atau pengobatan yang sesuai dengan jenis penyakit tulang yang diderita pasien. sistem ini dapat menyelesaikan masalah tertentu tanpa bantuan para ahli dalam bidang tersebut. Sedangkan bagi para ahli, sistem ini dapat digunakan untuk mendokumentasikan pengetahuan yang dimiliki. Setelah diuji dan dianalisis dengan melibatkan perhitungan secara manual, dapat diketahui bahwa secara garis besar hasil yang didapat dari perhitungan sistem sama dengan perhitungan manual. Sehingga secara umum sistem telah dapat memberikan solusi terhadap pengguna dan sistem berjalan sesuai dengan yang diharapkan.[5].

Menurut Turban dan Aronson sistem pakar memiliki dua bagian komponen utama, yaitu :

- a. Lingkungan pengembangan yang digunakan dalam sistem pakar untuk membangun komponen-komponennya dan menempatkan pengetahuan dalam basisnya.
- b. Lingkungan konsultasi yang digunakan oleh pemakai untuk mendapatkan pengetahuan dari pakar.

Lingkungan pengembangan digunakan sebagai sistem pakar baik dari segi pembangunan komponen maupun basis pengetahuan. Lingkungan konsultasi digunakan oleh seseorang yang bukan ahli untuk berkonsultasi.

Komponen-komponen yang ada pada sistem pakar , yaitu :

Fasilitas Akuisisi Pengetahuan

Akuisisi pengetahuan merupakan suatu proses untuk mengumpulkan data-data pengetahuan terhadap suatu masalah dari seorang pakar. Proses pengumpulan data pengetahuan ini dapat ditempuh dengan beberapa cara, yaitu mendapatkan pengetahuan dari buku, jurnal ilmiah, laporan dan wawancara langsung dengan pakar.

Basis Pengetahuan

Basis pengetahuan mengandung pengetahuan yang diperlukan untuk memahami, memformulasi dan menyelesaikan masalah. Basis pengetahuan pemecahan masalah dalam domain tertentu dan fakta-fakta tentang masalah. Basis pengetahuan ini didapatkan dari proses akuisisi pengetahuan. Basis pengetahuan ini direpresentasikan dalam bentuk tabel keputusan, yang akan menghasilkan kaidah produksi atau rule-rule yang akan diaplikasikan pada sistem.

Tabel Keputusan (*Decision Tables*)

Tabel keputusan adalah pengetahuan yang diorganisasikan dalam format *spreadsheet*, yaitu menggunakan kolom dan baris.

✚ Mesin Inferensi

Metode yang digunakan dalam mesin inferensi adalah *forward chaining*, yaitu penalaran yang didasarkan dengan mengumpulkan fakta-fakta (gejala-gejala) yang ada untuk menarik suatu kesimpulan [6]

Rule Based Reasoning merupakan suatu sistem pakar yang menggunakan aturan-aturan untuk menyajikan pengetahuannya. Sistem berbasis aturan adalah suatu perangkat lunak yang menyajikan keahlian pakar dalam bentuk aturan-aturan pada suatu domain tertentu untuk menyelesaikan suatu permasalahan. RBR adalah model sederhana yang bisa diadaptasi ke banyak masalah. Namun, jika aturan terlalu banyak, pemeliharaan sistem akan rumit dan terdapat banyak kesalahan dalam kerjanya.

Untuk membuat RBR, ada beberapa hal penting yang harus dimiliki:

1. Sekumpulan fakta yang mewakili *working memory*. Ini dapat berupa suatu keadaan yang relevan dengan keadaan awal sistem bekerja.
2. Sekumpulan aturan. Aturan ini mencakup setiap tindakan yang harus diambil dalam ruang lingkup permasalahan yang dibutuhkan.
3. Kondisi yang menentukan bahwa solusi telah ditemukan atau tidak (*non-exist*).

Hal ini berguna untuk menghindari *looping* yang tidak akan pernah berakhir. Teori RBR ini menggunakan teknik yang sederhana, dimulai dengan dasar aturan yang berisi semua pengetahuan dari permasalahan yang dihadapi yang kemudian dikodekan ke dalam aturan *if-then* yang mengandung data, pernyataan dan informasi awal. Sistem akan memeriksa semua aturan kondisi *if* yang menentukan subset, set konflik yang ada. Jika ditemukan, maka sistem akan melakukan kondisi *then*. Perulangan ini akan terus berlanjut hingga salah satu atau dua kondisi bertemu, jika aturan tidak diketemukan maka sistem tersebut harus keluar dari perulangan (*terminate*).

Untuk mengelola aturan, terdapat dua pendekatan yaitu:

1. *Forward Chaining* : aturan diproses berdasarkan sejumlah fakta yang ada, dan didapatkan konklusi sesuai dengan fakta-fakta tersebut. Pendekatan *forward chaining* disebut juga *data driven*.
2. *Backward Chaining* : diberikan target, kemudian aturan yang aksinya mengandung target di-*trigger*. *Backward chaining* ini cocok untuk menelusuri fakta yang masih belum lengkap, disebut juga *goal driven*.

Berikut ini beberapa kelebihan yang dimiliki oleh RBR :

a. Homogenitas

Karena memiliki sintaks yang seragam, makna dan interpretasi dari masing-masing aturan dapat dengan mudah dianalisis.

b. Kesederhanaan

Karena sintaks sederhana, mudah untuk memahami makna dari aturan. Ahli domain seringkali dapat memahami aturan tanpa penerjemahan yang eksplisit. Aturan sehingga dapat mendokumentasikan diri sampai batas yang baik.

c. Independensi

Ketika menambahkan pengetahuan yang baru tidak perlu khawatir tentang dimana aturan itu akan ditambahkan, atau apakah ada interaksi dengan aturan lainnya. Secara teori, setiap aturan adalah bagian independen dari pengetahuan tentang domain tersebut. Namun, dalam prakteknya, hal ini tidak sepenuhnya benar.

d. Modularitas

Independensi aturan mengarah ke modularitas dalam *rule base*. Prototipe sistem dapat diciptakan cukup cepat dengan membuat beberapa aturan. Hal ini dapat ditingkatkan dengan memodifikasi aturan berdasarkan kinerja dan menambahkan aturan baru.

Sedangkan beberapa kekurangan yang dimiliki oleh RBR antara lain:

- Jika terlalu banyak aturan, sistem menjadi sulit dalam memelihara *performance*.
- Keterbatasan dalam memutuskan teknik yang digunakan untuk suatu masalah.

Suatu Rule Terdiri dari 2 bagian, yaitu:

- a. Antecedent, yaitu bagian yang mengekspresikan situasi atau premis (Pernyataan berawalan IF)
- b. Konsekuen, yaitu bagian yang menyatakan suatu tindakan tertentu atau konklusi yang diterapkan jika situasi atau premis bernilai benar (Pernyataan berawalan THEN). Misalnya: IF lalulintas pagi ini macet THEN saya naik sepeda motor saja. Konsekuensi atau konklusi pada bagian THEN akan dinyatakan benar jika bagian IF pada sistem tersebut juga benar atau sesuai dengan aturan tertentu.

2. METODOLOGI PENELITIAN

Sedangkan Langkah-langkah dalam penelitian ini meliputi:

- Membuat analisis permasalahan :

Bagaimana sebuah *Group Decision Support System* (GDSS) akan dibangun untuk keperluan diagnosa suatu penyakit. Di dalam sebuah system GDSS biasanya terdiri dari beberapa pakar / dokter spesialis dibidangnya sebagai pengambil keputusan.

Kelompok pakar tersebut kita kelompokkan pada vector E dengan

$$E = \{e_1, \dots, e_r\} \quad (1)$$

Referensi yang diberikan untuk masing-masing alternatif kategori penyakit adalah

$$A = \{a_1, \dots, a_m\}$$

Setiap kategori penyakit biasanya memiliki kategori fitur- fitur tertentu seperti ; gejala, tanda atau ukuran, yang dilambangkan dengan C

$$C = \{c_1, \dots, c_n\} \quad (2)$$

- Membuat representasi dengan bilangan fuzzy linguistic :

Rancangan dari kriteria yang ada pada tahap analisis permasalahan akan lebih mudah jika dituangkan dalam bentuk tabel keputusan sebagai berikut :

Tabel 1. Tabel Keputusan hubungan antara fitur dengan katagori

Katagori (missal penyakit)	Fitur-fitur (mis. gejala, tanda atau ukuran)			
	C ₁	C ₂	...	C _n
A ₁	S ₁₁	S ₁₂		S _{1n}
A ₂	S ₂₁	S ₂₂		S _{2n}
...				
A _m	S _{m1}	S _{m2}		S _{mn}

Kemudian tabel keputusan tersebut diisi dengan nilai crisp (S_{ij}) bilangan fuzzy maupun bentuk linguistik

Setelah tabel keputusan terisi semuanya, kemudian bentuk linguistinya direpresentasikan dengan bilangan fuzzy segitiga $\check{A}=(u,\alpha,\beta)$ dengan derajat keanggotaan :

$$\mu_A[X] = \begin{cases} \frac{x-\alpha}{\mu-\beta} ; x \in [\alpha, \mu] \\ \frac{x-\beta}{\mu-\alpha} ; x \in [\mu, \beta] \\ 0 ; \text{lainnya} \end{cases} \quad (3)$$

Dimana α adalah batas atas dan β adalah batas bawah.

- Membuat matrik normalisasi :

Dari tabel keputusan tersebut, dapat dibentuk matrik X berukuran m x n, yang elemen-elemennya diambil dari elemen-elemen tabel keputusan yang terskala dalam rentang [0,1].

Misalkan tabel keputusan diatas untuk katagori C₁ dan C₂ diberikan nilai sbb :

Tabel 2 : Contoh Keputusan hubungan antara fitur dengan katagori Penyakit

Katagori Penyakit	Fitur-fitur	
	C ₁	C ₂
A ₁	Sewaktu-waktu	Lama
A ₂	Sering	Singkat

Kemudian diberikan bentuk linguistic untuk C₁ dan C₂ adalah sbb :

C₁ : sewaktu-waktu (0,25;0;0,5) ; jarang = (0,5;0,25;0,75) sedangkan sering (0,75;0,5;1)

C₂ : singkat (0,25;0;0,5) ; cukup = (0,5;0,25;0,75) dan lama (0,75;0,5;1)

Maka dapat dibuat matrik x yang mersesuaian dengan tabel keputusan diatas adalah :

$$X = \begin{bmatrix} 0,25 & 0,75 \\ 0,75 & 0,25 \end{bmatrix} \quad (4)$$

- Membuat representasi linguistik untuk setiap pakar

Representasi linguistik dari setiap pakar berupa vektor

$$L^k = \{1_1^k, 1_2^k, \dots, 1_m^k\} \quad (5)$$

Dengan 1_1^k adalah representasi linguistik terhadap kategori penyakit ke-i. Setelah itu dilakukan penyeragaman format preferensi fuzzy. Salah satu kegunaan dari transformasi ini adalah untuk melakukan penyeragaman format preferensi, apabila proses pengambilan keputusannya didasarkan pada banyak pakar (Group).

Misalkan diberikan 2 alternatif S_i dan S_j secara linguistik

$$\check{A}_i = (u_i, \alpha_i, \beta_i) \text{ dan } \check{A}_j = (u_j, \alpha_j, \beta_j) \quad (6)$$

Maka akan digunakan fungsi defuzzy Max g dan transformasi f, sebagai berikut (leekwijck, 1999)(Zhou,2000)

$$P_{ij}^k = g\left(f(\check{A}_i, \check{A}_j)\right) = \frac{u_i^2}{u_i^2 + u_j^2} \quad (7)$$

- Menentukan bobot setiap fitur dan perengkingan

Nilai rengking tiap kategori penyakit dirumuskan dengan

$$d_1 = \sum_{j=1}^n b_{1j} w_j \quad (8)$$

- Implementasi dengan *Rule Based Reasoning* (RBR)

Nilai d_i yang selanjutnya akan digunakan sebagai nilai kepercayaan pada aturan dalam system pakar *Rule Based Reasoning* (RBR) yang akan disimpan dalam basis pengetahuan dari system pakar tersebut.

3. Hasil dan Pembahasan

Hubungan antara penyakit dengan gejalanya akan direpresentasikan dengan bilangan fuzzy segitiga $\check{A} = (\mu, \alpha, \beta)$

Contoh :

Di dalam system pakar diketahui Rule base sbb :

R1 : IF C1 AND C2 AND C3 AND C4 AND C5 AND C6 THEN penyakit AAAA

R2 : IF C1 AND C2 AND C3 AND C4 AND C5 AND C6 THEN penyakit BBBB

R3 : IF C1 AND C2 AND C3 AND C4 AND C5 AND C6 THEN penyakit CCCC

R4 : IF C1 AND C2 AND C3 AND C4 AND C5 AND C6 THEN penyakit DDDD

Bentuk linguistic untuk hubungan penyakit dan gejalanya :

- Tidak pernah
- Pernah
- Kadang-kadang
- Sering
- Sangat sering

Seperti yang terlihat pada gambar berikut :

The screenshot shows a software window titled "Form Hubungan Penyakit & Gejala". It contains a text input field for "No. Kasus". Below it is a section titled "TABEL KEPUTUSAN" which is a table with the following structure:

Kategori Penyakit	Fitur-fitur (gejala, tanda, atau ukuran)					
	C1	C2	C3	C4	C5	C6
A1	[dropdown]	[dropdown]	[dropdown]	[dropdown]	[dropdown]	[dropdown]
A2	tidak pernah pernah kadang-kadang sering sangat sering	[dropdown]	[dropdown]	[dropdown]	[dropdown]	[dropdown]
A3	[dropdown]	[dropdown]	[dropdown]	[dropdown]	[dropdown]	[dropdown]
A4	[dropdown]	[dropdown]	[dropdown]	[dropdown]	[dropdown]	[dropdown]

Below the table, there is a "Keterangan" section with two lines: "A : Kategori Penyakit (Misalkan : Migran, Hipertensi)" and "C : Gejala / Tanda-tanda Penyakit (Misalkan : frekuensi sakit, kualitas rasa sakit)". A "proses" button is located at the bottom right.

Gambar 1. form hubungan antara penyakit dengan gejala yang direfresentasikan dalam bentuk linguistic

kemudian setelah semua hubungan antara penyakit (A) dengan gejala (C) terisi maka akan menghasilkan sebuah matrik seperti terlihat pada tampilan berikut :

The screenshot shows a software window titled "Form Data Penyakit". It contains a text input field for "No_Rule" with the value "R5". Below it is a section titled "TABEL KEPUTUSAN" which displays a linguistic matrix. The matrix is shown as:

Diagnosa Penyakit	G1	Matrik				G4
T1	Sedang	0.75	0.8	0.8	0.95	Tinggi
T2	Sedang	0.75	0.8	0.9	0.825	Sedang
T3	Sedang	0.75	0.95	0.8	0.7	Rendah
T4	Sedang	0.75	0.8	0.8	0.7	Rendah

Below the matrix, it says "Rule : R5" and has "Simpan" and "Lanjut" buttons. A "proses" button is at the bottom right.

Gambar 2. Matrik linguistic hubungan antara penyakit dengan gejala-nya

Matrik tersebut akan disimpan terlebih dahulu, kemudian langkah berikutnya adalah menentukan bobot pada tiap gejala pada masing-masing penyakit.

Kemudian dengan menggunakan Utility vector to fuzzy preference relation (Chiclana, 1998) :

$$P_{ij}^k = \frac{(\mu_i^k)^2}{(\mu_i^k)^2 + (\mu_j^k)^2} ; 1 \leq i \neq j \leq m$$

Setelah semua nilai matrik didapatkan maka langkah selanjutnya adalah :Menjumlahkan elemen-elemen matriknya sbb :

$$I1 = 0 + 0.692 + 0.9 + 0.692 = 2.284$$

$$I2 = 1.608$$

Dan seterusnya. Kemudian nilai-nilai dari I1 sampai I6 tersebut disusun dari yang terbesar ke yang terkecil. Barulah setelah itu bobot akan didapatkan Seperti pada tampilan berikut :

The screenshot shows a software window titled "Preferensi Pakar". It contains several input fields and two tables. On the left, there are dropdown menus for "No Rule" (set to R5), "Kode Pakar" (set to P1), and "Referensi Fuzzy Gejala Penyakit" with four items: C1 (Tinggi), C2 (Cukup), C3 (Rendah), and C4 (Cukup). In the center, a "Matriks Perbandingan" table is displayed with the following values:

0	0.692	0.9	0.692
0.308	0	0.8	0.5
0.1	0.2	0	0.2
0.308	0.5	0.8	0

On the right side, there are buttons for "Matrik", "Bobot", "Simpan", and "Lanjut". Below the matrix, a "Bobot" table shows the calculated weights:

W1	W2	W3	W4
0.617	0.188	0.152	0.043

Gambar 3. bobot dari masing-masing gejala penyakit

Untuk nilai bobot dari pakar yang kedua caranya sama dengan langkah-langkah sebelumnya.

4. KESIMPULAN

Penerapan metode *fuzzy linguistic multi-expert* ini dapat membantu menentukan nilai ketidakpastian dari sistem pakar yang dibangun hal ini dikarenakan jika kita menggunakan nilai ketidakpastian pada setiap rule maka kita akan mengalami kesulitan untuk mendapatkannya dari seorang pakar dalam bidang tertentu. Sedangkan metode ini nilai ketidakpastiannya diperoleh dengan cara memasukkan nilai gejala atau evidence yang menggunakan inputan fuzzy linguistic setelah itu baru direngkingkan. Nilai perengkingan pada metode ini sangat dipengaruhi oleh inputan nilai dari gejala, dimana nilainya nanti akan digunakan sebagai nilai ketidakpastian pada rule based dalam sistem pakar.

DAFTAR PUSTAKA

- [1]. Turban, E., Aronson, J.E., 2005, "*Decision Support System and Intelligent System*", 5 ed, Prentice Hall inc., USA
- [2]. Kwok, Ron C.W., Ma, Jian dan Zhou, Duanning, 2005, "*Improving Group Decision Making ;Fuzzy GSS Approach* ", Dept of Information System University of Hong Kong
- [3]. Kusumadewi, Sri, 2003, "*Artificial Intelligence(Teknik dan Aplikasi)*", Graha Ilmu, Yogyakarta
- [4]. Judea Pearl, 1991, "*Probabilistic Reasoning in Intelligent Systems*", Morgan Kaufmann Publishers, San Mateo California, USA
- [5]. Heckerman, D.,1990, "*Probabilistic Interpretations for Mycin's Certainty Factors*", Elsevier Science Publishers B.V, North-Holland
- [6]. Giarratano, dan Riley G., 1994. "*Expert System: Principle and Progaming*", 2 ed, PWS Publishing Company, Boston
- [7]. Firebaugh, M. W., 1989, "*Artificial Intelligence Knowledge-Based Approach*", PWS-KENT Publishing Company, Boston