

## Penerapan Metode Topsis Dalam Sistem Pakar Diagnosa Penyakit Vertigo

**Rudy Hartono, Utin Kasma**

STMIK Pontianak; Jl. Merdeka Barat no. 372, (0561) 735555  
Jurusan Teknik Informatika, STMIK Pontianak, Pontianak  
e-mail: [rdxhartz@gmail.com](mailto:rdxhartz@gmail.com), [utin.kasma@stmikpontianak.ac.id](mailto:utin.kasma@stmikpontianak.ac.id)

### **Abstrak**

*Vertigo merupakan suatu gejala dengan sensasi diri sendiri atau sekeliling terasa berputar yang terjadi secara tiba-tiba. Sering kali orang yang terkena vertigo tidak mengetahui diri sedang mengalami gejala dari vertigo, banyak yang beranggapan itu hanya sebagai pusing kepala biasa. Oleh karena itu penelitian ini bertujuan untuk membuat suatu aplikasi yang dapat berguna untuk dapat mendeteksi seseorang apakah terkena gejala dari vertigo atau tidak. Bentuk penelitian adalah menggunakan studi kasus, yaitu dengan mempelajari gejala dari penyakit Vertigo. Sistem pakar ini menggunakan metode Topsis untuk metodenya. Penggunaan metode TOPSIS dalam merancang aplikasi dimaksudkan karena konsep dari TOPSIS sederhana dan mudah dipahami, komputasinya efisien, dan memiliki kemampuan mengukur kinerja relatif dari alternatif-alternatif keputusan dalam bentuk matematis yang sederhana. Di dalam perancangan sistem pakar ini, penulis menggunakan aplikasi Visual Basic.Net dan database PhpMyAdmin dalam membuat sistem pakar ini Hasil penelitian ini berupa sebuah aplikasi yang dapat digunakan untuk mendiagnosa penyakit vertigo dari gejala-gejala yang akan ditanyakan oleh sistem kepada user yang akan diproses oleh sistem untuk dapat memberikan jawaban kepada user sesuai gejala-gejala yang dialami oleh user.*

**Kata kunci**— *Vertigo, Sistem Pakar, Php Myadmin, Topsis*

### **Abstract**

Vertigo is a symptom with a sensation of oneself or round was spinning that occurs suddenly. Often people affected by vertigo does not know himself is experiencing symptoms of vertigo, many consider it only as an ordinary headache. Therefore, this study aims to create an application that could be useful to be able to detect a person if they develop symptoms of vertigo or not. Form of research is the use of case studies, by studying the symptoms of the disease Vertigo. This expert system uses TOPSIS method to method. The use of TOPSIS method in designing applications intended for the concept of TOPSIS simple and easy to understand, efficient computation, and has the ability to measure the relative performance of the alternatives in the decision of a simple mathematical form. In the expert system design, the author uses Visual Basic.Net and database PhpMyAdmin in making this expert system results of this study in the form of an application that can be used to diagnose diseases of the vertigo symptoms that will be asked by the system to the user that will be processed by system to be able to provide answers to a user regarding the symptoms experienced by the user.

**Keywords**— *Vertigo, expert system, Php Myadmin, Topsis*

## 1. PENDAHULUAN

Penyakit Vertigo merupakan keadaan dimana seseorang merasa dirinya atau ruangan disekitarnya seolah berputar. Perasaan ini disebabkan oleh virus, penurunan aliran darah ke otak, trauma kepala, iskemi sementara karena obat-obatan dan *alcohol*, *migraine* dan beberapa kasus *epilepsy*. Vertigo bisa berlangsung hanya beberapa saat atau bisa berlanjut sampai beberapa jam bahkan bilangan hari. Penyakit Vertigo termasuk dalam kategori penyakit menegah atau tidak terlalu ringan juga, karena apabila penyakit vertigo ini menyerang maka penderitanya tidak dapat melakukan apa pun karena gejala vertigo mengganggu sehingga sulit untuk melakukan beragam aktivitas yang sedang dilakukan.

Dalam mendiagnosa penyakit Vertigo ini para penderitanya harus memeriksakan diri kepada dokter syaraf untuk mengetahui apakah si penderita terkena penyakit vertigo atau tidak. Masalah yang dihadapi oleh masyarakat kebanyakan penderita penyakit Vertigo tidak mengetahui apakah dia terkena penyakit Vertigo itulah sebabnya dalam penelitian ini membuat sebuah sistem kecerdasan buatan untuk dapat membantu, mempermudah, dan memberikan hasil yang konsisten dalam membuat keputusan .

Kecerdasan buatan merupakan salah satu bagian ilmu komputer yang membuat agar mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan manusia . Sistem pakar (*Expert system*) secara umum adalah sebuah sistem yang berusaha untuk mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh manusia . Sistem Pakar dapat mengumpulkan dan menyimpan pengetahuan seorang pakar atau beberap orang pakar dalam komputer. Pengetahuan tersebut kemudian akan digunakan oleh pengguna untuk menyelesaikan suatu masalah. Sistem pakar dapat membantu dalam memberikan solusi dari masalah yang ada .

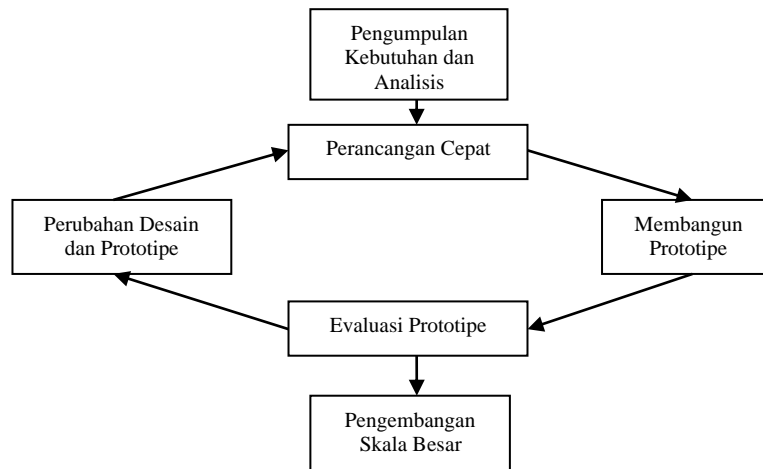
Dalam penelitian ini implementasi sistem pakar akan diterapkan dalam bidang kesehatan yaitu untuk mendiagnosa penyakit vertigo pada manusia. Sistem pakar ini diharapkan dapat memberikan informasi mengenai penyakit vertigo dan dapat membantu dalam melakukan diagnosa melalui gejala – gejala yang ditimbulkan dari penyakit vertigo. Dalam penelitian ini akan digunakan matriks / bobot tetap pada bagian konvolusi, sehingga dapat mengurangi penggunaan sumber daya komputasi yang ada. Penggunaan bobot yang terlatih pada bagian konvolusi dalam algoritma jaringan saraf tiruan konvolusi mengizinkan sistem untuk bekerja dengan performa yang lebih baik dan cepat mencapai titik konvergen Dalam penelitian ini metode yang dipakai adalah metode TOPSIS.

TOPSIS merupakan salah satu metode yang dapat digunakan untuk menyelesaikan masalah *Multi Attribute Decision Making* (MADM) secara praktis dan lebih efisien. Penggunaan metode *TOPSIS* pada penelitian kali ini dimaksudkan karena metode *TOPSIS* memiliki bobot dalam mempertimbangkan gejala sehingga metode ini sudah hampir mirip dengan pemikiran seorang pakar untuk mendiagnosa penyakit. Sehingga metode ini cocok untuk diterapkan untuk mendiagnosa penyakit Vertigo karena dalam perhitungan *TOPSIS* mempertimbangkan bobot dalam menentukan hasilnya sehingga hasilnya lebih akurat.

Penggunaan metode *TOPSIS* pada penelitian kali ini dimaksudkan karena metode *TOPSIS* memiliki bobot dalam mempertimbangkan gejala sehingga metode ini sudah hampir mirip dengan pemikiran seorang pakar untuk mendiagnosa penyakit. Sehingga metode ini cocok untuk diterapkan untuk mendiagnosa penyakit Vertigo karena dalam perhitungan *TOPSIS* mempertimbangkan bobot dalam menentukan hasilnya. Sehingga hasilnya lebih akurat dan memiliki tingkat error yang lebih kecil.

## 2. METODE PENELITIAN

Dalam penelitian ini, bentuk penelitian yang dipakai adalah studi kasus (*case study*), yaitu salah satu metode penelitian yang bertujuan untuk mencari secara intensif dari latar belakang dan kejadian sekarang terhadap suatu objek yang diteliti yang dijadikan untuk sebuah kasus dengan menggunakan cara-cara yang sistematis yang dilakukan dengan melihat mengamati apa yang akan dibutuhkan dan metode penelitian *Reserch and Development*. Metode perancangan perangkat lunak menggunakan *Prototype*. Metode prototipe memperjelas spesifikasi kebutuhan yang diperlukan oleh sistem kepada pengembang.



Gambar 1 Diagram Langkah – Langkah Metode Prototipe

Pada gambar 1 menjelaskan tentang metode pengujian yang akan digunakan adalah metode pengujian *blackbox*, yaitu pengujian yang memungkinkan perekayasa perangkat lunak mendapatkan serangkaian input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pembuatan aplikasi ini menggunakan bahasa pemrograman Visual Basic.

*Technique for Order Preference by Similiarity to Ideal Solution* (TOPSIS) merupakan salah satu metode yang dapat digunakan untuk menyelesaikan masalah *Multi Atribute Decision Making* (MADM) secara praktis. Hal ini disebabkan karena konsep dari TOPSIS sederhana dan mudah dipahami, komputasinya efisien, dan memiliki kemampuan mengukur kinerja relative dari alternatif-alternatif keputusan dalam bentuk matematis yang sederhana.

Adapun langkah-langkah perhitungan dalam metode TOPSIS adalah (Desi Leha Kurniasih, 2013:8-10):

- a. Menentukan jenis-jenis kriteria
- b. Menentukan ranking setiap alternatif pada setiap kriteria
- c. Membangun matriks keputusan
- d. Menentukan bobot preferensi untuk setiap kriteria
- e. Membangun *normalized decision matrix*
- f. Elemen  $r_{ij}$  hasil dari normalisasi *decision matrix* R dengan metode *Euclidean length of a vector* adalah (dalam rumus 4.1):

$$r_{ij} = \frac{X_{ij}}{\sqrt{\sum_i^m X_{ij}^2}} \dots\dots\dots (1)$$

Dimana :

$r_{ij}$  = hasil dari normalisasi matriks keputusan R

$$i = 1, 2, 3, \dots, m;$$

$$j = 1, 2, 3, \dots, n;$$

- g. Membangun *weighted normalized decision matrix*

Dengan bobot  $W = (w_1, w_2, \dots, w_n)$ , maka normalisasi bobot matriks  $V$  adalah (dalam rumus 4.2) :

$$V = \begin{bmatrix} w_{11}r_{11} & \dots & w_{1n}r_{1n} \\ \vdots & \ddots & \vdots \\ w_{m1}r_{m1} & \dots & w_{nm}r_{nm} \end{bmatrix} \dots\dots\dots (2)$$

- h. Menentukan solusi ideal positif dan solusi ideal negative

Solusi ideal positif dinotasikan dengan  $A^+$  dan solusi ideal negative dinotasikan dengan  $A^-$ , sebagai berikut (dalam rumus 4.3 dan 4.4):

Menentukan solusi (+) dan (-)

$$A^+ = \{(max_{vij})(min_{vij} | j \in j'), i = 1, 2, 3, \dots, m\} = \{V_1^+, V_2^+, \dots, V_m^+\} \quad (3)$$

$$A^- = \{(min_{vij})(max_{vij} | j \in j'), i = 1, 2, 3, \dots, m\} = \{V_1^-, V_2^-, \dots, V_m^-\} \quad (4)$$

Dimana :

$V_{ij}$  = elemen matriks  $V$  baris ke- $i$  dan kolom ke- $j$

$j = \{j=1, 2, 3, \dots, n \text{ dan } j \text{ berhubungan dengan } \textit{benefit criteria}\}$

$j' = \{j=1, 2, 3, \dots, n \text{ dan } j \text{ berhubungan dengan } \textit{cost criteria}\}$

- i. Menghitung separasi

*Separation measure* ini merupakan pengukuran jarak dari suatu alternative ke solusi ideal positif dan solusi ideal negatif. Perhitungan matematisnya adalah sebagai berikut (dalam rumus 4.5 dan 4.6):

*Separation measure* untuk solusi ideal positif

$$S_i^+ = \sqrt{\sum_{j=1}^n (V_{ij} - V_j^+)^2}, \text{ dengan } i = 1, 2, 3, \dots, m \dots\dots\dots (5)$$

*Separation measure* untuk solusi ideal negative

$$S_i^- = \sqrt{\sum_{j=1}^n (V_{ij} - V_j^-)^2}, \text{ dengan } i = 1, 2, 3, \dots, m \dots\dots\dots (6)$$

- j. Menghitung kedekatan relatif terhadap solusi ideal. Kedekatan relatif dari alternative  $A^+$  dengan solusi ideal  $A^-$  direpresentasikan dengan :

$$C_i = \frac{S_i^-}{S_i^- + S_i^+}, \text{ dengan } 0 < C_i^+ \text{ dan } i = 1, 2, 3, \dots, m \dots\dots\dots (7)$$

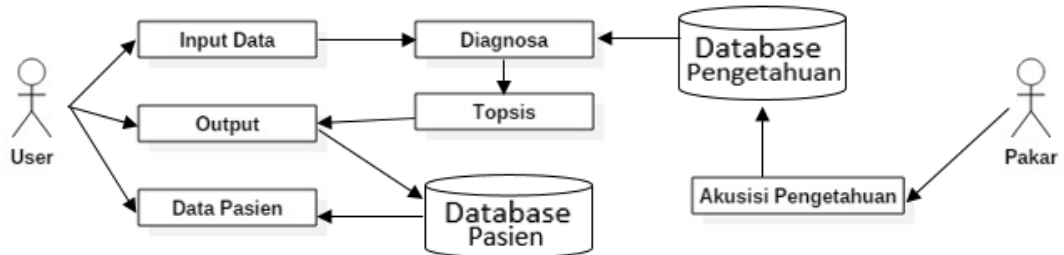
- k. Merangking alternative

Alternatif dapat dirangking berdasarkan urutan  $C_i^+$ . Maka dari itu, alternatif terbaik adalah salah satu yang berjarak terpendek dengan solusi ideal negatif.

Untuk metode pengujiannya adalah menggunakan pengujian blackbox yaitu pengujian memungkinkan perakayasa perangkat lunak mendapatkan serangkaian input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pembuatan aplikasi menggunakan Visual Basic.NET dengan basis data MySQL

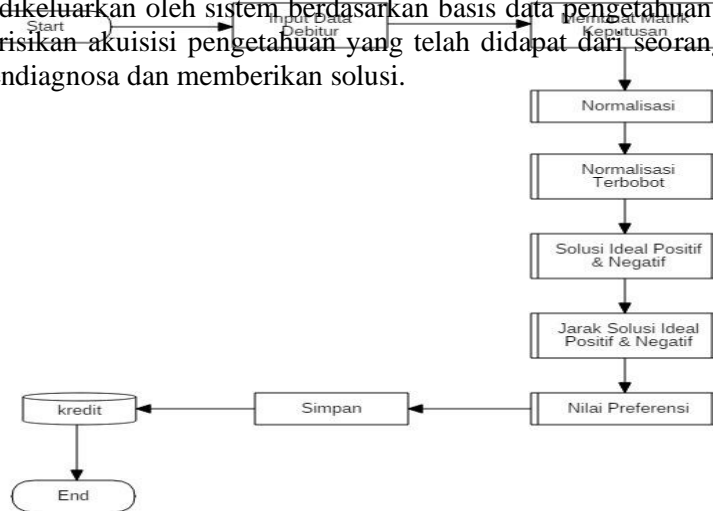
### 3. HASIL DAN PEMBAHASAN

Terdapat beberapa bagian pada arsitektur sistem pakar diagnosa vertigo ini diantaranya (Gambar 4).



Gambar 2 Arsitektur Sistem Pakar Diagnosa Vertigo

Pada gambar 2 memberikan penjelasan pada masing – masing bagian yang terdapat dalam sistem memperlihatkan gambaran umum dari sistem pakar diagnosa Vertigo. Gambar arsitektur di atas menunjukkan *user* menggunakan sistem pakar dan memberikan *input* yang sesuai dengan pertanyaan dari sistem pakar. Dari *input* yang diberikan *user*, sistem pakar mulai mendiagnosa, dimana sistem akan melihat pola diagnosa yang sesuai. Diagnosa penyakit yang dilakukan sistem pakar adalah dengan menggunakan metode *topsis* untuk melihat dan mempelajari pola-pola sudah pernah diujikan untuk memperbaiki bobot yang ada. Setelah diagnosa penyakit selesai, selanjutnya hasil diagnosa dan solusi atas penyakit yang diderita oleh pengguna akan dikeluarkan oleh sistem berdasarkan basis data pengetahuan. Dimana basis data pengetahuan berisikan akuisisi pengetahuan yang telah didapat dari seorang pakar atau dokter syaraf untuk mendiagnosa dan memberikan solusi.

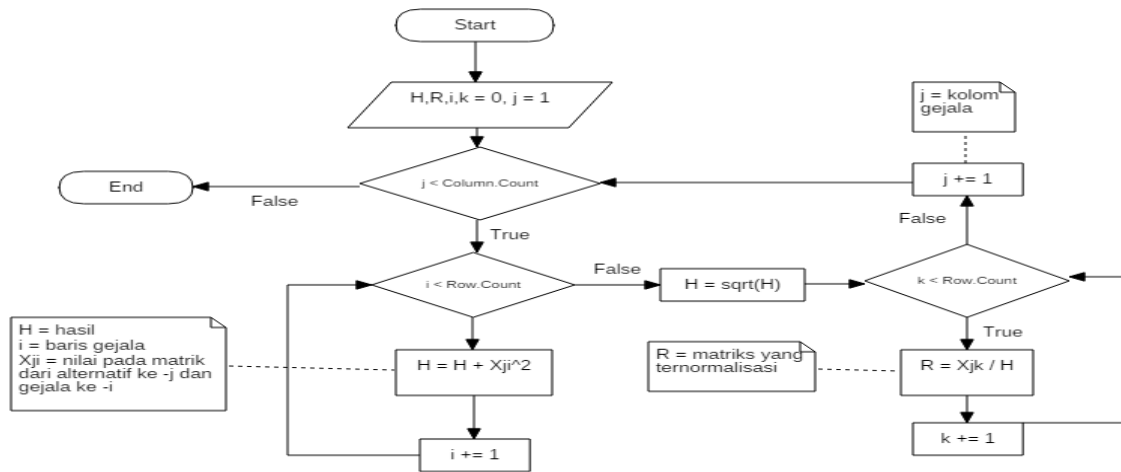


Gambar 3 Alur Perhitungan Topsis

Gambar 3 menjelaskan tentang alur utama dari sistem yang akan digunakan untuk mendapat hasil yang terbaik. Alur tersebut berupa dari gejala menuju untuk membuat matrik keputusan setelah sistem langsung menuju normalisasi dan selanjutnya menuju normalisasi terbobot dan mencari nilai solusi ideal positif dan negatif dan akan mendapatkan nilai jarak solusi ideal positif dan negatif, setelah itu akan masuk kepada tahap terakhir berupa mencari nilai preferensi dan nilai preferensi sistem

akan menyimpannya kedalam database. .Sebelum itu data tersebut akan dnormalisasi untuk mendapatkan hasil yang ternormalisasi dari data yang telah dimasukkan kedalam sistem yang nanti setelah dari proses ternormalisasi itu untuk selanjutnya akan dilanjutkan untuk menghitung dari nilai yang sudah dinormalisasi tersebut .

Selanjutnya akan ada proses ternormalisasi yang bertujuan untuk mendapatkan suatu nilai yang akan digunakan untuk menghitung sistem tersebut (Gambar 4).



Gambar 4 Flowchart Ternormalisasi

Gambar 4 menjelaskan tentang proses perhitungan hasil yang akan dinormalisasikan terlihat jelas alur tersebut setelah memulai sistem, sistem akan masuk kedalam perhitungan normalisasi dapat terlihat dengan jelas proses alur perhitungan pada sistem tersebut dari perhitungan matriks yang akan dinormalisasi sampai kepada hasil akhir yang berupa nilai yang telah dinormalisasikan.

Alternatif yang digunakan pada pengambilan keputusan ini adalah Jenis penyakit Vertigo yang terbagi menjadi 5 jenis yaitu Vertigo Sentral, Vertigo Periferal, Vertigo Nonvestibular, Vertigo Vestibular dan dapat dilihat pada table 1 sebagai berikut :

Id_penyakit	Penyakit
P1	Vertigo Sentral
P2	Vertigo Periferal
P3	Vertigo Nonvestibular
P4	Vertigo Vestibular

Tabel 1 Tabel Penyakit Vertigo

Gejala yang digunakan dalam aplikasi kali ada sebanyak 23 gejala yang meliputi gejala-gejala penyakit Vertigo yang ada seperti dibawah ini, dalam perhitungan ini hanya menampilkan beberapa contoh perhitungan :

## Penerapan Metode Topsis Dalam Sistem Pakar Diagnosa Penyakit Vertigo

G1 = Sulit berdiri.

G2 = Lemas (tidak mampu menggenggam erat benda dengan telapak tangan).

G3 = Sukar menelan.

G4 = Keadaan sekitar terasa berputar, benda diam di sekitar nampak bergoyang (Oscillpsia) dan nampak ganda (Diplopia).

G5 = Kelumpuhan otot-otot wajah.

Berdasarkan dengan data diatas maka diberikan bobot yang bernilai 0 -5 yang bertujuan untuk memberikan nilai bobot kepada gejala-gejala yang diajukan penjelasan nilai bobot akan ditampilkan pada tabel 2:

Bobot Keputusan	Nilai
Menggambarkan intensitas yang dirasakan oleh user yang mempunyai skala 0 yang berarti paling rendah dan 5 menggambarkan skala paling tinggi	5
	4
	3
	2
	1
	0

Tabel 2 Bobot Keputusan

Berikut adalah contoh dari bobot keputusan yang aka dihitung :

<b>Id_pass</b>	<b>Nama_pasien</b>	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>Nama_penyakit</b>
<b>A001</b>	<b>ERVI</b>	2	2	3	2	4	<b>Vertigo Sentral</b>
<b>A002</b>	<b>DEWI</b>	3	4	2	4	4	<b>Vertigo Vestibular</b>
<b>A003</b>	<b>SHINTA</b>	5	3	4	1	4	<b>Vertigo Non_vestibular</b>

<b>A004</b>	<b>ANFI</b>	3	4	2	3	5	<b>Vertigo Periferal</b>
<b>A005</b>	<b>DANI</b>	1	3	5	2	5	<b>Vertigo Vestibular</b>

Tabel 3 Contoh Bobot Keputusan

Setelah melihat data dari tabel, data akan diproses dengan normalisasi yang ada seperti dibawah ini :

Membuat matriks keputusan yang ternormalisasi berdasarkan persamaan berikut:

$$|X1| = \sqrt{(2^2) + (3^2) + (5^2) + (3^2) + (1^2)} = 6.928$$

$$R11 = \frac{X11}{|X1|} = \frac{2}{6.928} = 0.288$$

$$R12 = \frac{X12}{|X1|} = \frac{3}{6.928} = 0.433$$

$$R13 = \frac{X13}{|X1|} = \frac{5}{6.928} = 0.721$$

$$R14 = \frac{X14}{|X1|} = \frac{3}{6.928} = 0.433$$

$$R15 = \frac{X15}{|X1|} = \frac{1}{6.928} = 0.144$$

$$|X2| = \sqrt{(2^2) + (4^2) + (3^2) + (4^2) + (3^2)} = 7.348$$

$$R21 = \frac{X21}{|X2|} = \frac{2}{7.348} = 0.272$$

$$R22 = \frac{X23}{|X2|} = \frac{4}{7.348} = 0.544$$

$$R23 = \frac{X23}{|X2|} = \frac{3}{7.348} = 0.408$$

$$R24 = \frac{X24}{|X2|} = \frac{4}{7.348} = 0.544$$

$$R25 = \frac{X25}{|X2|} = \frac{3}{7.348} = 0.408$$

Demikian seterusnya sehingga diperoleh matriks ternormalisasi R:

$$R = \begin{bmatrix} 0.288 & 0.272 & 0.393 & 0.342 & 0.404 \\ 0.433 & 0.544 & 0.262 & 0.685 & 0.404 \\ 0.721 & 0.408 & 0.525 & 0.171 & 0.404 \\ 0.433 & 0.544 & 0.262 & 0.514 & 0.505 \\ 0.144 & 0.408 & 0.656 & 0.342 & 0.505 \end{bmatrix}$$

Membuat matriks keputusan yang ternormalisasi terbobot (matriks Y) dengan mengkalikan bobot  $W_i$  dengan rating kerja  $R_{ij}$  berdasarkan persamaan berikut:

$$\begin{aligned}
 Y_{11} &= W_1 * R_{11} = (2) * (0.288) = 0.577 \\
 Y_{12} &= W_2 * R_{12} = (4) * (0.272) = 1.088 \\
 Y_{13} &= W_3 * R_{13} = (5) * (0.393) = 1.969 \\
 Y_{14} &= W_4 * R_{14} = (3) * (0.342) = 1.028 \\
 Y_{15} &= W_5 * R_{15} = (2) * (0.404) = 0.808 \\
 Y_{21} &= W_1 * R_{21} = (2) * (0.433) = 0.866 \\
 Y_{22} &= W_2 * R_{22} = (4) * (0.544) = 2.177
 \end{aligned}$$

Demikian seterusnya sehingga diperoleh matriks Y:

$$R = \begin{bmatrix} 0.577 & 1.088 & 1.969 & 1.028 & 0.808 \\ 0.866 & 2.177 & 1.313 & 2.057 & 0.808 \\ 1.443 & 1.632 & 2.626 & 0.514 & 0.808 \\ 0.866 & 2.177 & 1.313 & 1.543 & 1.01 \\ 0.288 & 1.632 & 3.282 & 1.028 & 1.01 \end{bmatrix}$$

Menentukan matriks solusi ideal positif (A+) dan matriks solusi ideal negative (A-) berdasarkan rating bobot ternormalisasi  $Y_{ij}$ . Solusi ideal positif (A+) dihitung berdasarkan persamaan berikut:

$$\begin{aligned}
 Y_{1+} &= \min \{0.577, 0.866, 1.443, 0.866, 0.288\} = 0.288 \\
 Y_{2+} &= \max \{1.088, 2.177, 1.632, 2.177, 1.632\} = 2.177 \\
 Y_{3+} &= \max \{1.969, 1.313, 2.626, 1.313, 3.282\} = 3.282 \\
 Y_{4+} &= \max \{1.028, 2.057, 0.514, 1.543, 1.028\} = 2.057 \\
 Y_{5+} &= \max \{0.808, 0.808, 0.808, 1.01, 1.01\} = 1.01
 \end{aligned}$$

Solusi ideal negatif (A-) dihitung berdasarkan persamaan berikut:

$$\begin{aligned}
 Y_{1-} &= \max \{0.577, 0.866, 1.443, 0.866, 0.288\} = 1.443 \\
 Y_{2-} &= \min \{1.088, 2.177, 1.632, 2.177, 1.632\} = 1.088 \\
 Y_{3-} &= \min \{1.969, 1.313, 2.626, 1.313, 3.282\} = 0.674 \\
 Y_{4-} &= \min \{1.028, 2.057, 0.514, 1.543, 1.028\} = 0.514 \\
 Y_{5-} &= \min \{0.808, 0.808, 0.808, 1.01, 1.01\} = 1.01
 \end{aligned}$$

Keterangan : jika kriteria bersifat “cost” maka A+ mencari nilai terkecil (min) sedangkan A- mencari nilai terbesar (max)

Menentukan jarak antara nilai setiap alternatif dengan matriks solusi ideal positif dan matriks ideal negative. Jarak antara nilai terbobot setiap alternatif terhadap solusi ideal positif, dihitung berdasarkan persamaan berikut:

$$\begin{aligned}
 D_{1+} &= \sqrt{(0.288 - 0.577)^2 + (2.177 - 1.088)^2 + (3.282 - 1.969)^2 + (2.057 - 1.028)^2 + (1.01 - 0.808)^2} \\
 &= 2.012 \\
 D_{2+} &= \sqrt{(0.288 - 0.866)^2 + (2.177 - 2.177)^2 + (3.282 - 1.313)^2 + (2.057 - 2.057)^2 + (1.01 - 0.808)^2} \\
 &= 2.052 \\
 D_{3+} &= \sqrt{(0.288 - 1.443)^2 + (2.177 - 1.632)^2 + (3.282 - 2.626)^2 + (2.057 - 0.514)^2 + (1.01 - 0.808)^2} \\
 &= 2.107
 \end{aligned}$$


---

---

$D4+$

$$= \sqrt{(0.288 - 0.866)^2 + (2.177 - 2.177)^2 + (3.282 - 1.313)^2 + (2.057 - 1.543)^2 + (1.01 - 1.01)^2}$$

$$= 2.125$$

$D5+$

$$= \sqrt{(0.288 - 0.288)^2 + (2.177 - 1.632)^2 + (3.282 - 3.282)^2 + (2.057 - 1.028)^2 + (1.01 - 1.01)^2}$$

$$= 1.181$$

Jarak antara nilai terbobot setiap alternative terhadap solusi ideal negatif, dihitung berdasarkan persamaan sebagai berikut:

$D1-$

$$= \sqrt{(1.443 - 0.577)^2 + (1.088 - 1.088)^2 + (1.313 - 1.969)^2 + (0.514 - 1.028)^2 + (0.808 - 0.808)^2}$$

$$= 1.219$$

$D2-$

$$= \sqrt{(1.443 - 0.866)^2 + (1.088 - 2.177)^2 + (1.313 - 1.313)^2 + (0.514 - 2.057)^2 + (0.808 - 0.808)^2}$$

$$= 1.985$$

$D3-$

$$= \sqrt{(1.443 - 1.443)^2 + (1.088 - 1.632)^2 + (1.313 - 2.626)^2 + (0.514 - 0.514)^2 + (0.808 - 0.808)^2}$$

$$= 1.435$$

$D4-$

$$= \sqrt{(1.443 - 0.866)^2 + (1.088 - 2.177)^2 + (1.313 - 1.313)^2 + (0.514 - 1.543)^2 + (0.808 - 1.01)^2}$$

$$= 1.605$$

$D5-$

$$= \sqrt{(1.443 - 0.288)^2 + (1.088 - 1.632)^2 + (1.313 - 3.282)^2 + (0.514 - 1.028)^2 + (0.808 - 1.01)^2}$$

$$= 2.402$$

Menentukan nilai preferensi untuk setiap alternative ( $V_i$ )

Kedekatan setiap alternative terhadap solusi ideal dihitung berdasarkan persamaan berikut:

$$V1 = \frac{1.219}{2.102 + 1.219} = 0.377$$

$$V2 = \frac{1.985}{1.985 + 2.052} = 0.491$$

$$V3 = \frac{1.435}{1.435 + 2.107} = 0.405$$

$$V4 = \frac{1.605}{1.605 + 2.125} = 0.43$$

$$V5 = \frac{2.402}{2.402 + 1.181} = 0.6703$$

Berdasarkan hasil perhitungan yang telah dilakukan, dapat diketahui bahwa  $V5$  memiliki nilai terbesar. Dari Perhitungan ini kita dapat melihat nilai bobot yang terbesar yang nanti nilai bobot terbesar itu adalah hasil yang akan menentukan hasil akhir dari proses perhitungan ini yang nantinya akan digunakan untuk memberikan solusi atau nilai preferensi yang diinginkan yang sesuai dengan cara perhitungan yang digunakan.

#### 4. KESIMPULAN

Hasil sistem pakar diagnosa penyakit Vertigo ini menghasilkan kesimpulan bahwa sistem pakar ini dibangun menggunakan metode *Topsis* dan dalam pengembangan aplikasinya menggunakan Visual Basic.Net. Sistem pakar ini dapat mendiagnosa jenis penyakit Vertigo berdasarkan gejala-gejalanya yang dialami oleh pengguna yang digunakan *user* untuk

---

melakukan diagnosa gejala-gejala penyakit Vertigo tanpa harus bertemu dengan seorang pakar yang berguna bagi pengguna yang mengalami gejala-gejala dari penyakit Vertigo untuk dapat mengecek apakah pengguna terkena penyakit Vertigo atau hanya gejalanya saja.

### 5. SARAN

Sistem harus dapat melakukan diagnosa terhadap gejala-gejala penyakit vertigo menggunakan metode topsis, sistem harus bisa dikembangkan lagi untuk bisa mendiagnosa penyakit vertigo agar memperoleh hasil yang lebih akurat serta memiliki tingkat error yang lebih kecil menggunakan metode lain atau dengan menggabungkan dengan metode yang lainnya.

### DAFTAR PUSTAKA

- [1] Arhami, Muhammad. 2005. *Konsep Dasar Sistem Pakar*. Yogyakarta: Andi.
- [2] Arikunto, Suharsimi. 2006. *Metodelogi penelitian*. Yogyakarta: Bina Aksara
- [3] Hasibuan, Zainal A., 2007, *Metodologi Penelitian Pada Bidang Ilmu Komputer dan Teknologi Informasi*, Depok: Fakultas Ilmu Komputer Universitas Indonesia.
- [4] Kadir, Abdul, 2008, *Tuntunan Praktis Belajar Database Menggunakan MySQL*, C.V Andi Offset, Yogyakarta.
- [5] Kusumadewi, Sri. 2003. *Arficial Intelligence(Teknik dan Aplikasinya)*. Yogyakarta:Graha Ilmu.
- [6] Salahuddin M, (2011), *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*, Bandung: Modula.
- [7] Onelin, Training Solutions. 2001. *Microsoft Access Version 2002: Step By Step*. Jakarta: Elex Media Komputindo.
- [8] S., Rosa A dan Shallahuddin, M., 2013, *Rekayasa Perangkat Lunak Terstruktur, Teori dan Prakterk Aplikasi Bisnis, Andi, Yogyakarta*.
- [9] Sugiyono. 2003. *Metode Penelitian Bisnis*. Bandung: Pusat Bahasa Depdiknas.
- [10] Turban, Efraim., E.Aronson, Jay., Liang, Ting-Peng, 2005, *Decision Support Systems and Intelligent System*, Yogyakarta: Andi.
- [11] Shalahudin, A.S. Rosa., 2013, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*, Penerbit Informatika, Bandung.
- [12] Pressman, S. Roger, 2012, *Rekayasa Perangkat Lunak Pendekatan Praktisim*, edisi 7, C.V ANDI OFFSET, Yogyakarta.