

Case-Based Reasoning untuk Diagnosa Penyakit THT (Telinga Hidung dan Tenggorokan)

Tedy Rismawan^{*1}, Sri Hartati²

¹Prodi S2/S3 Ilmu Komputer dan Elektronika, FMIPA UGM

²Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: *tedyrismawan@yahoo.com, shartati@ugm.ac.id

Abstrak

Case-Based Reasoning (CBR) merupakan sistem penalaran komputer yang menggunakan pengetahuan lama untuk mengatasi masalah baru. CBR memberikan solusi terhadap kasus baru dengan melihat kasus lama yang paling mendekati kasus baru. Hal ini akan sangat bermanfaat karena dapat menghilangkan kebutuhan untuk mengekstrak model seperti yang dibutuhkan oleh sistem berbasis aturan. Penelitian ini mencoba untuk membangun suatu sistem Penalaran Berbasis Kasus untuk melakukan diagnosa penyakit THT (Telinga, Hidung dan Tenggorokan). Proses diagnosa dilakukan dengan cara memasukkan kasus baru (target case) yang berisi gejala-gejala yang akan didiagnosa ke dalam sistem, kemudian sistem akan melakukan proses indexing dengan metode backpropagation untuk memperoleh indeks dari kasus baru tersebut. Setelah memperoleh indeks, sistem selanjutnya melakukan proses perhitungan nilai similarity antara kasus baru dengan basis kasus yang memiliki indeks yang sama menggunakan metode cosine coefficient. Kasus yang diambil adalah kasus dengan nilai similarity paling tinggi. Jika suatu kasus tidak berhasil didiagnosa, maka akan dilakukan revisi kasus oleh pakar. Kasus yang berhasil direvisi akan disimpan ke dalam sistem untuk dijadikan pengetahuan baru bagi sistem. Hasil penelitian menunjukkan sistem penalaran berbasis kasus untuk mendiagnosa penyakit THT ini membantu paramedis dalam melakukan diagnosa. Hasil uji coba sistem terhadap 111 data kasus uji, terdapat 9 kasus yang memiliki nilai similarity di bawah 0.8.

Kata kunci—*case-based reasoning, indexing, similarity, backpropagation, cosine coefficient*

Abstract

Case-Based Reasoning (CBR) is a reasoning system that uses old knowledge to solve new problem. CBR provides solutions to new cases by looking at old case that comes closest to the new case. It will be very useful because it eliminates the need to extract the model as required by the rule-based systems. This study tries to establish a system for CBR for diagnosing diseases of ENT. Diagnosis process is done by inserting a new case that contains the symptoms of the disease to be diagnosed, then the system will do the indexing process with backpropagation method to obtain an index of new cases. After that, the system will do the calculation of the value of similarity between the new case by case basis which has the same index with new cases using cosine coefficient method. The case taken is the case with the highest similarity value. If a case is not successfully diagnosed, the case will be revised by the experts and it can be used as new knowledge for the system. The results showed case-based reasoning system to diagnose diseases of ENT can help paramedics in performing diagnostics. The test results of 111 data test cases, obtained 9 cases that have similarity values below 0.8.

Keywords—*case-based reasoning, indexing, similarity, backpropagation, cosine coefficient*

1. PENDAHULUAN

Case-based reasoning (CBR) merupakan sistem penalaran yang menggunakan pengetahuan lama untuk menyelesaikan permasalahan baru. Penyajian pengetahuan (*knowledge representation*) dibuat dalam bentuk kasus-kasus (*cases*). Setiap kasus berisi masalah dan jawaban, sehingga kasus lebih mirip dengan suatu pola tertentu. Cara kerja CBR adalah dengan membandingkan kasus baru dengan kasus lama, jika kasus baru tersebut mempunyai kemiripan dengan kasus lama maka CBR akan memberikan jawaban kasus lama untuk kasus baru tersebut. Jika tidak ada yang cocok maka CBR akan melakukan adaptasi dengan memasukkan kasus baru tersebut ke dalam database penyimpanan kasus (*case base*), sehingga secara tidak langsung pengetahuan CBR akan bertambah. Kelebihan utama dari CBR dibandingkan dengan sistem berbasis aturan (*rule base system*) adalah dalam hal akuisisi pengetahuan, dimana pada sistem CBR dapat menghilangkan kebutuhan untuk ekstrak model atau kumpulan dari aturan-aturan, seperti yang diperlukan dalam model/sistem yang berbasis aturan.

Jika kasus lama yang telah ada pada basis kasus (*case base*) memiliki jumlah yang banyak, maka akan muncul kendala dalam lamanya waktu yang diperlukan untuk memperoleh hasil dari proses pencarian kasus yang mirip karena sistem harus menghitung nilai kemiripan kasus baru terhadap semua kasus lama yang telah ada. Oleh karena itu, diperlukan proses *indexing*, yaitu proses pengelompokkan kasus yang ada berdasarkan fitur yang ditentukan. Proses *indexing* dilakukan agar pada saat proses pencarian kasus yang mirip pada kasus baru sistem CBR cukup menghitung nilai kemiripan terhadap kasus yang ada pada kelompok yang sama dengan kasus baru tersebut. Proses *indexing* dapat dilakukan dengan menggunakan metode pengelompokkan data. Salah satu metode pengelompokkan yang dapat digunakan adalah *backpropagation* (perambatan balik). *Backpropagation* merupakan metode dalam *Artificial Neural Network* (ANN) yang dapat melakukan pengelompokkan terhadap vektor *input*. Inti dari algoritma pembelajaran dengan metode *backpropagation* terletak pada kemampuannya untuk mengubah nilai-nilai bobotnya untuk menanggapi adanya error. Perubahan bobot ditujukan untuk meminimalkan jumlah kesalahan jaringan. Jumlah kesalahan akan semakin mengecil dengan berjalannya perulangan yang dilakukan oleh jaringan tersebut. Proses mencari kedekatan atau kemiripan antara kasus baru dengan kasus lama untuk memperoleh solusi terhadap kasus baru pada CBR dapat menggunakan berbagai macam metode, dimana metode ini akan mempengaruhi keberhasilan dari CBR. Salah satu metode yang dapat digunakan dalam mencari kedekatan atau kemiripan kasus baru dengan kasus lama adalah *cosine coefficient*. Metode ini merupakan metode yang digunakan untuk mengukur kemiripan dari 2 vektor dengan cara mengukur nilai kosinus sudut dari kedua vektor tersebut. *Cosine coefficient* merupakan pengukuran *similarity* yang paling sering digunakan untuk keperluan *clustering* pada dokumen maupun aplikasi temu kembali informasi (*information retrieval*) pada umumnya [1].

Jumlah penyakit pada domain THT cukup banyak, begitu pula dengan kasus-kasus penyakit THT yang pernah terjadi. Hal ini mengakibatkan munculnya kebutuhan akan sebuah sistem yang dapat membantu pakar maupun paramedis untuk melakukan diagnosa terhadap pasien berdasarkan gejala yang dialami. Sistem tersebut harus dapat melakukan pengelompokkan terhadap kasus baru, karena banyaknya kasus yang disimpan sebagai pengetahuan, sehingga proses pencarian kemiripan kasus baru dapat dilakukan hanya terhadap kasus-kasus pada basis kasus yang memiliki kelas yang sama dengan kasus baru. Selain itu, sistem juga harus dapat menghitung nilai *similarity* kasus baru terhadap semua kasus yang memiliki indeks yang sama dengan kasus baru tersebut.

Berdasarkan hal tersebut, penulis bermaksud untuk melakukan penelitian berkaitan dengan diagnosa penyakit THT yang diimplementasikan dalam bentuk CBR dengan menggunakan metode *backpropagation* dan *cosine coefficient*. Diharapkan penelitian ini dapat menghasilkan suatu sistem yang dapat membantu melakukan proses diagnosa secara cepat dan tepat.

Penelitian yang berfokus pada *case base reasoning* banyak sekali ditemui dengan berbagai macam metode dan berbagai macam bentukan kasus yang ada dalam bidang *science* pada sistem cerdas (*intelligent systems*). Salem, dkk[2] membangun suatu *prototype case-based expert system* untuk membantu diagnosis penyakit jantung. Sebanyak 110 buah kasus telah dikoleksi untuk 4 jenis penyakit jantung yaitu: mitral stenosis, left-sided heart failure, stable angina pectoris dan essential hypertension. Masing-masing kasus berisikan 207 atribut mengenai demografis dan data klinis. Setelah memindahkan duplikasi kasus, sistem melatih 42 kasus untuk pasien penyakit jantung Egyptian. Analisa statistik dilakukan untuk menentukan nilai-nilai penting dari ciri-ciri kasus. Dua macam metode *retrieval* yang digunakan yaitu: metode induksi dan nearest-neighbor. Hasil mengindikasikan bahwa *retrieval* dengan metode nearest neighbor lebih baik dibandingkan metode induksi, dimana akurasi *retrieval* berturut-turut adalah 100% dan 53,8%. Ahli jantung telah mengevaluasi seluruh performa sistem ini dimana sistem mampu memberikan suatu hasil diagnosa yang benar untuk 13 kasus baru.

Kim dan Han[3] membahas tentang penggunaan *self-organizing maps* (SOM) dan *learning vector quantization* (LVQ) pada proses *indexing* kasus untuk penalaran berbasis kasus. Penelitian yang dilakukan membandingkan hasil *indexing* dengan metode induktif dan SOM serta LVQ, sedangkan proses perhitungan *similarity* menggunakan metode nearest neighbor dengan pengukuran jarak Euclidian. Hasil penelitian yang dilakukan memperoleh hasil bahwa CBR dengan proses *indexing* memiliki performa yang lebih tinggi dari CBR tanpa *indexing*, dan perbandingan dari metode induksi, SOM dan LVQ memperoleh hasil bahwa metode yang terbaik adalah LVQ, kemudian SOM dan terakhir adalah metode induksi.

Farhan dkk [4] membahas tentang penggunaan metode hashing untuk melakukan proses *indexing* pada *case base*. Penelitian ini menggunakan data kasus dari operasional bendungan Timah Tasoh pada tahun 1997-2005. Hasil dari penelitian ini menunjukkan bahwa proses *retrieval* dengan *indexing* menggunakan metode hashing memberikan hasil yang lebih baik daripada penggunaan metode CBR secara konvensional, dimana proses *retrieval* dengan menggunakan hashing sebagai metode *indexing* rata-rata memiliki kecepatan proses lebih cepat 1 ms.

2. METODE PENELITIAN

2.1 Deskripsi Sistem

Proses pada sistem dimulai dengan melakukan pembentukan indeks untuk kasus-kasus yang ada, indeks pada kasus yang ada diperoleh dari pakar. Setelah diperoleh indeks dari setiap kasus yang ada, maka proses selanjutnya adalah melatih kasus-kasus yang ada dengan backpropagation, dimana gejala dari kasus menjadi data input dan indeks pada kasus menjadi target. Hasil dari pelatihan backpropagation berupa nilai bobot akhir yang nantinya akan digunakan untuk proses *indexing* pada kasus baru. Proses selanjutnya yang terjadi pada sistem adalah menginisialisasi gejala yang dialami oleh pasien yang dianggap sebagai kasus baru, setelah itu sistem akan melakukan proses *indexing* terhadap kasus baru tersebut berdasarkan nilai bobot akhir backpropagation yang telah dilatih sebelumnya berdasarkan kasus-kasus lama. Setelah memperoleh indeks dari kasus baru, maka selanjutnya sistem melakukan perhitungan nilai *similarity* kasus baru terhadap kasus yang lama yang memiliki indeks sama. Proses perhitungan *similarity* menggunakan Cosine Coefficient. Nilai *similarity* berkisar antara 0 sampai dengan 1. Apabila *similarity* kasus baru dengan salah satu kasus yang ada pada basis kasus bernilai 1, berarti kasus baru tersebut sama dengan kasus lamayang ada dalam basis kasus. Apabila *similarity* kasus baru memiliki nilai 0.8, maka kasus baru akan menggunakan solusi yang sama dengan kasus lama yang ada pada basis kasus. Namun, apabila nilai *similarity* tidak mencapai nilai 0.8, maka dianggap kasus baru tersebut tidak memiliki solusi dan kasus tersebut selanjutnya akan disimpan sebagai kasus baru yang nantinya akan dievaluasi oleh pakar (*revise*)

dan disimpan kembali ke dalam sistem sebagai kasus baru dengan solusi yang telah diberikan (*retain*).

2.2. Representasi Kasus

Tabel 1 menunjukkan hubungan antara gejala dengan penyakit. Pada tabel tersebut terdapat 5 gejala yaitu G1 hingga G5 dan 5 penyakit yaitu P1 hingga P5. Selain itu, juga terdapat 1 field yang berisi nilai kelas (indeks) dari masing-masing penyakit yang diberi label K. Pada penelitian ini, terdapat 3 macam kelas utama yang membagi kasus, yaitu kelas untuk kasus Telinga yang diwakili dengan K=1, kelas untuk kasus Hidung yang diwakili dengan K=2 dan kelas untuk kasus Tenggorok yang diwakili dengan K=3. Kelas pada masing-masing kasus berpengaruh pada sistem, karena proses diagnosa pada sistem akan dibagi menjadi 3 kelas tersebut. Selain dibagi menjadi 3 kelas utama tersebut, kasus juga akan dibagi menjadi 9 sub kelas (SK) berdasarkan wilayah serangan dari penyakit yang dimiliki masing-masing kasus. Masing-masing kelas memiliki jumlah sub kelas yang berbeda, yaitu:

- Kasus dengan indeks utama telinga memiliki 2 subkelas, yaitu: Otitis, dan Tuba Eustachii.
- Kasus dengan indeks utama hidung memiliki 3 subkelas, yaitu: Rinitis, Sinus, dan bagian hidung lainnya.
- Kasus dengan indeks utama tenggorok memiliki 4 subkelas, yaitu: Tenggorok bagian atas, Faring dan Fauisum, Orofaring dan Tonsil. Sub kelas inilah yang nantinya akan digunakan sebagai target pada proses indeksing.

Tabel 1 Representasi kasus

	G1	G2	G3	G4	G5	K	SK
P1	1	0	0.33	1	0	1	1
P2	1	0.67	0	1	1	1	2
P3	0.67	0	0.67	0	0	2	3
P4	0	0.67	0	1	1	3	4
P5	0.33	0	0	0	1	3	4

Nilai yang terdapat pada Tabel 1 merupakan nilai yang merepresentasikan pengaruh dari gejala terhadap masing-masing penyakit (kecuali nilai pada field K). Nilai tersebut terbagi menjadi 4 jenis, yaitu:

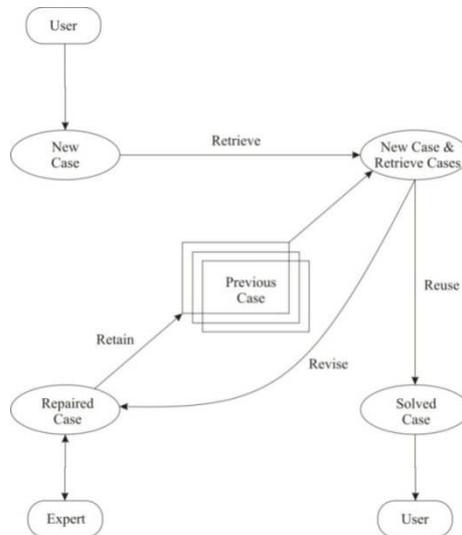
- 0 yang berarti gejala tersebut tidak dirasakan.
- 0.33 yang berarti gejala tersebut dirasakan namun hanya sedikit.
- 0.67 yang berarti gejala tersebut cukup dirasakan.
- 1 yang berarti gejala tersebut sangat dirasakan.

2.3 Diagram Blok

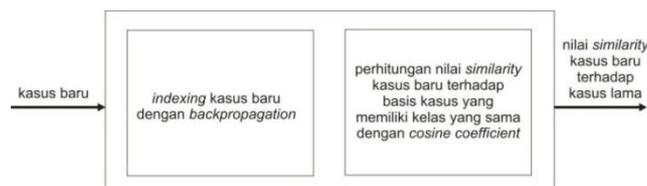
Diagram blok yang dapat dilihat pada Gambar 1 merupakan gambaran sistem secara konseptual. Diagram blok mengikuti arsitektur CBR *cycle* yaitu *retrieve*, *reuse*, *revise* dan *retain* serta dengan beberapa penyesuaian berdasarkan dengan sistem yang dikembangkan. *User* yang bertindak sebagai pemakai sistem membuat *target case* atau kasus baru yaitu dengan cara memasukkan gejala-gejala yang dialami, kemudian sistem akan melakukan *indexing* terhadap kasus baru. Setelah memperoleh indeks dari kasus tersebut kemudian dilakukan perhitungan untuk mencari nilai *similarity* antara kasus baru tersebut dengan *source case* atau kasus-kasus yang ada dalam *case base* yang memiliki indeks yang sama. Proses ini dapat dilihat pada Gambar 2.

Proses perbandingan tadi akan menghasilkan sebuah nilai yang dapat menentukan ada atau tidak kemiripan antara *target case* dengan *source case*, jika ada kemiripan maka solusi dari *source case* akan diberikan kepada *user*. Revisi kasus akan dilakukan jika tidak ada kemiripan antara *target case* dengan *source case* setelah melakukan konfirmasi terlebih dahulu dengan

pakar. Setelah pakar melakukan konfirmasi kasus yang direvisi, kasus tersebut kemudian disimpan ke dalam *case base* untuk menjadi pengetahuan baru (*retain*).



Gambar 1 Diagram blok sistem penalaran berbasis kasus



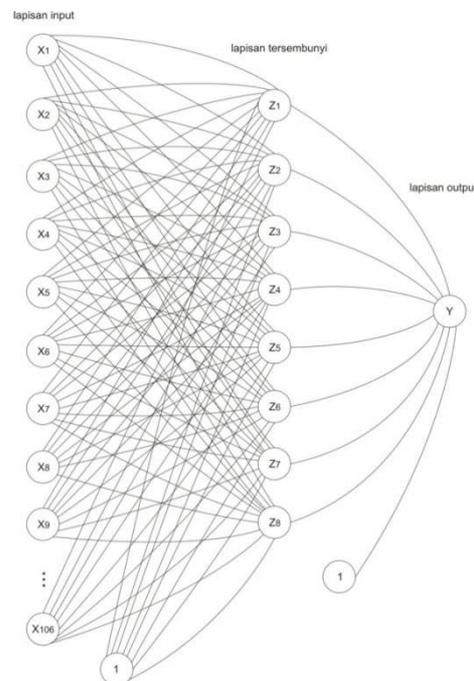
Gambar 2 Diagram blok proses *retrieval*

2.4 Indexing

Indexing merupakan proses pengelompokkan kasus yang ada berdasarkan fitur yang ditentukan. Proses *indexing* dapat mengefisienkan waktu dan memory karena pada saat melakukan proses pencarian kasus yang mirip dengan kasus yang baru, sistem CBR tidak perlu menghitung nilai kemiripan kasus baru terhadap semua kasus yang ada, namun cukup menghitung nilai kemiripan terhadap kasus yang berada pada kelompok yang sama dengan kasus baru tersebut. Pada penelitian ini metode yang digunakan untuk *indexing* adalah *backpropagation*.

Backpropagation merupakan algoritma pembelajaran yang terawasi dan biasanya digunakan oleh perceptron dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron-neuron yang ada pada lapisan tersembunyinya. Algoritma *backpropagation* menggunakan error output untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan error ini, tahap perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu [5].

Proses *indexing* dengan *backpropagation* terbagi menjadi 2 bagian, yaitu proses pelatihan dan proses pengujian. Proses pelatihan dimulai dengan memasukkan data gejala dari kasus lama sebagai data *input* dan kelas pada masing-masing kasus tersebut menjadi target pelatihan. Gambar 3 merupakan arsitektur dari jaringan *backpropagation* yang digunakan pada penelitian ini, dimana terdapat 106 unit input jaringan (X_1 sampai dengan X_{106}), dimana nilai ini diperoleh berdasarkan jumlah gejala ada. Jaringan *backpropagation* yang digunakan memiliki 1 lapisan tersembunyi yang memiliki unit sebanyak 8 (Z_1 sampai dengan Z_8), serta 1 unit output yang merupakan hasil akhir dari jaringan *backpropagation* (Y).



Gambar 3 Arsitektur backpropagation yang digunakan

2.5 Retrieval dan Similarity

Teknik retrieval yang digunakan dalam penelitian ini adalah teknik nearest neighbor. Nearest neighbor adalah pendekatan mencari kasus dengan menghitung kedekatan antara kasus baru dengan kasus lama. Ide dasar dari teknik ini adalah membandingkan setiap atribut-atribut target case dengan atribut-atribut source case yang ada dalam case base, kemudian perbandingan tersebut dihitung dengan menggunakan fungsi similarity. Pada penelitian ini digunakan Cosine Coefficient sebagai fungsi similarity. Nilai batas similarity yang digunakan pada penelitian ini adalah 0.8, sehingga apabila pada proses konsultasi diperoleh nilai similarity di bawah 0.8 maka kasus tersebut belum memiliki solusi.

Cosine Coefficient Similarity merupakan metode untuk mengukur kemiripan antara 2 buah vektor dengan cara mengukur nilai kosinus sudut dari kedua vektor tersebut. Jika diberikan 2 buah vektor X dan Y, maka nilai kesamaannya dapat dicari dengan persamaan:

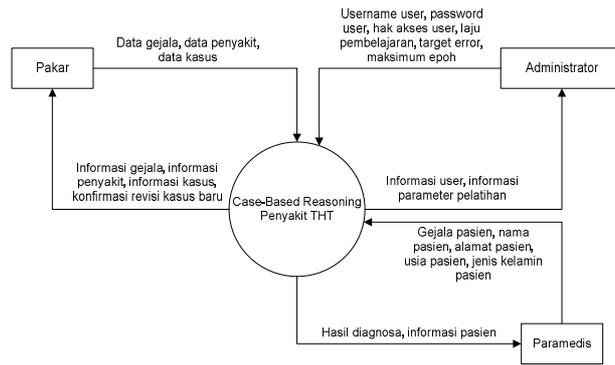
$$\cos(\angle X, Y) = \frac{\langle X, Y \rangle}{\|X\| \|Y\|} \quad (1)$$

dimana “ $\langle \rangle$ ” menunjukkan perkalian dari kedua vektor, dan “ $\| \|$ ” menunjukkan *norm* pada masing-masing vektor. Untuk vektor dengan elemen-elemen non-negatif, nilai kemiripan kosinus selalu terletak antara 0 dan 1, dimana 1 menunjukkan kedua vektor tersebut benar-benar sama, sedangkan 0 menunjukkan sebaliknya [6].

2.6 Desain Sistem

Diagram Context

Gambar 4 merupakan diagram context yang menggambarkan tingkat tertinggi dari sistem secara keseluruhan.

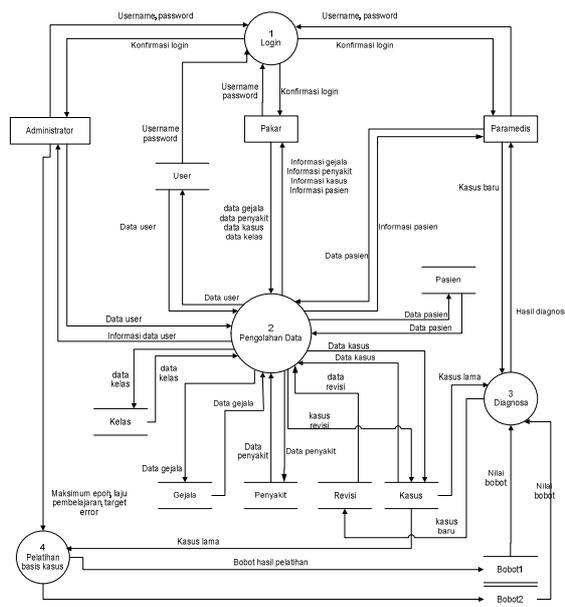


Gambar 4 Diagram context

Diagram context terdiri dari 3 elemen lingkungan yaitu Administrator, Pakar dan Paramedis serta sebuah proses secara keseluruhan dari sistem *case-based reasoning* untuk mendiagnosa penyakit THT. Administrator akan memberikan *input* ke sistem berupa data *user* (pengguna sistem) dan parameter pelatihan. Parameter pelatihan digunakan untuk pelatihan jaringan *backpropagation* pada saat proses *indexing*. Pakar akan memberikan *input* ke sistem berupa data penyakit, data gejala penyakit, kasus untuk case base serta melakukan revisi dan retain dari suatu kasus. Data penyakit dan data gejala penyakit merupakan data-data master yang diperlukan oleh sistem untuk bekerja. Paramedis akan memberikan *input* ke sistem berupa data pasien dan kasus baru, dimana kasus tersebut berisi gejala-gejala penyakit yang akan didiagnosa. Jika kasus tersebut berhasil didiagnosa, maka Paramedis akan menerima hasil diagnosa dan menyimpan data pasien yang memiliki kasus tersebut. Apabila kasus baru tidak berhasil didiagnosa, maka proses revisi akan dilakukan oleh Pakar, dan hasil dari revisi akan disimpan sebagai basis kasus pada sistem. Administrator, Pakar dan Paramedis harus melakukan proses login terlebih dahulu untuk dapat mengakses sistem.

DFD Level 1

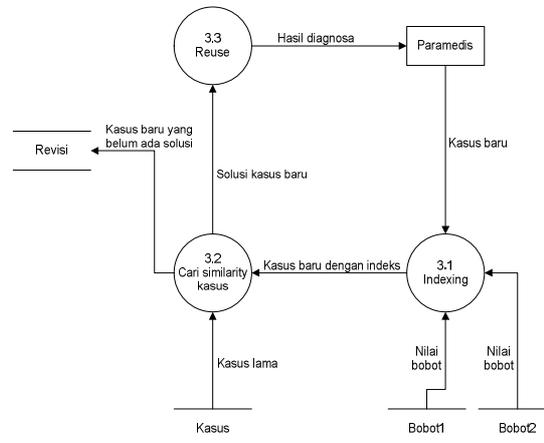
Pemecahan diagram context akan menghasilkan DFD level 1 yang dibagi menjadi 4 sub-sistem utama, yaitu sub-sistem login, sub-sistem pengolahan data, sub-sistem diagnosa dan sub-sistem pelatihan basis kasus. Gambar 5 merupakan gambaran aliran data yang terjadi pada DFD level 1.



Gambar 5 DFD level 1

DFD Level 2 Proses Diagnosa

Sedangkan aliran data pada proses diagnosa dapat dilihat pada gambar 6 yang merupakan DFD level 2 proses diagnosa

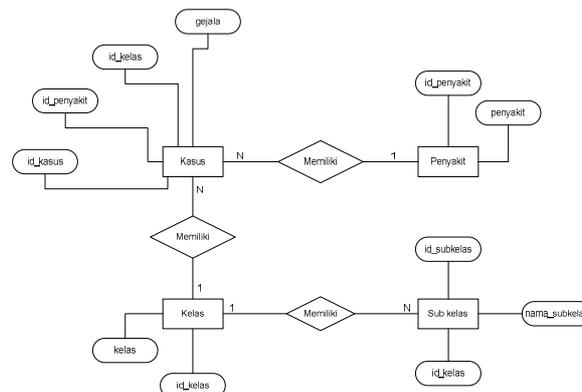


Gambar 6 DFD level 2 proses diagnosa

Rancangan basis data

Perancangan basis data pada sistem ini menggunakan beberapa aturan bisnis (*business rules*). Berikut ini merupakan aturan bisnis yang digunakan pada sistem CBR untuk Penyakit THT:

- Setiap kasus memiliki 1 penyakit
- Satu penyakit dapat digunakan oleh beberapa kasus
- Setiap kasus memiliki 1 kelas
- Satu kelas dapat digunakan oleh beberapa kasus
- Satu kelas dapat memiliki beberapa sub kelas
- Setiap sub kelas hanya dapat digunakan oleh 1 kelas.



Gambar 7 Entity relationship diagram

2.7 Implementasi Sistem

Tahapan penggunaan sistem dimulai dengan melakukan pelatihan terhadap basis kasus terlebih dahulu, agar diperoleh nilai bobot yang nantinya akan dipergunakan pada saat melakukan *indexing* terhadap kasus baru. Setelah proses pelatihan dilakukan, maka diperoleh bobot akhir dari jaringan yang akan digunakan pada proses *indexing* kasus baru. Gambar 9 merupakan tampilan sistem untuk halaman konsultasi pasien. Pada halaman ini terdapat beberapa bagian utama yang berisi gejala-gejala, yaitu halaman bagian gejala umum, bagian gejala telinga, bagian gejala hidung dan bagian gejala tenggorok. Untuk melakukan diagnosa, user harus memilih jenis dan tingkatan gejala pada sistem sesuai yang dirasakan.

Gambar 9 Halaman konsultasi bagian umum

3. HASIL DAN PEMBAHASAN

3.1 Proses pengisian case base

Tahap awal dari penggunaan sistem proses adalah pengisian *case base*. Data-data kasus yang akan dimasukkan ke dalam *case base* diperoleh dari data rekam medis Poliklinik Telinga Hidung Tenggorok (THT) RSUD dr. Soedarso, Pontianak. Terdapat 106 gejala THT dan 38 nama penyakit yang dibagi menjadi 3 kelas dan 9 subkelas. Kasus yang dimasukkan ke dalam *case base* sebanyak 208 kasus.

Setiap gejala penyakit dibagi menjadi 4 kategori yaitu Tidak, Sedikit, Cukup dan Ya. Nilai untuk masing-masing kategori adalah 0 untuk Tidak, 0.33 untuk Sedikit, 0.67 untuk Cukup dan 1 untuk Ya. Pemberian nilai dari masing-masing gejala untuk setiap kasus dilakukan oleh Pakar.

3.2 Proses pelatihan basis kasus

Proses pelatihan data kasus dengan *backpropagation* dilakukan untuk memperoleh bobot jaringan yang akan digunakan pada proses *indexing* kasus baru. Pada proses ini jaringan *backpropagation* digunakan untuk melatih semua data yang ada pada basis kasus dengan data gejala sebagai *input* pelatihan dan data subkelas sebagai target pelatihan.

Sebelum melakukan pelatihan, admin sebagai *user* yang memiliki hak akses proses pelatihan harus mengisi parameter pelatihan terlebih dahulu. Pemilihan parameter jaringan akan mempengaruhi pembelajaran yang dilakukan oleh jaringan, sehingga harus digunakan parameter yang baik dalam melakukan pelatihan. Namun, untuk memperoleh parameter pelatihan yang baik maka harus dilakukan percobaan satu per satu terhadap parameter jaringan. Pada penelitian ini parameter pelatihan yang digunakan adalah:

- Laju pembelajaran : 0.3
- Target error : 0.00001
- Maksimum epoch : 15000

3.3 Hasil pengujian

Pada bagian ini akan dilakukan pengujian kasus *riil* terhadap sistem. Kasus yang akan diuji memiliki gejala sebagai berikut:

- Demam : Ya
- Nyeri telinga : Ya
- Rasa penuh di telinga : Ya
- Pendengaran berkurang : Cukup
- Telinga berdengung : Sedikit
- Cairan telinga bening : Ya

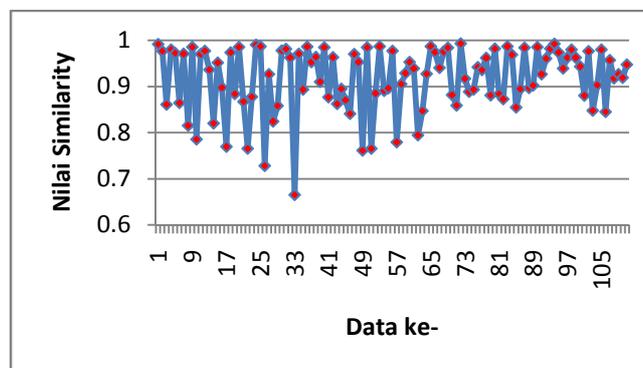
Nilai *similarity* dengan *Cosine Coefficient* dihitung berdasarkan persamaan (1), dimana kasus uji dihitung nilai *similarity*-nya satu per satu terhadap 41 kasus yang memiliki indeks yang sama. Setelah nilai *similarity* dari kasus uji dihitung terhadap semua kasus yang memiliki indeks sama maka diperoleh nilai *similarity* sebanyak 41 nilai. Tabel 3 menunjukkan nilai *similarity* yang diperoleh dari perhitungan *similarity* dengan *Cosine Coefficient* antara kasus uji dengan kasus yang pada basis kasus yang memiliki indeks yang sama.

Tabel 3 Nilai *similarity* kasus uji dengan kasus yang memiliki indeks sama

Kasus ke-	Nilai <i>similarity</i>	Kasus ke-	Nilai <i>similarity</i>	Kasus ke-	Nilai <i>similarity</i>
1	0.9561	15	0.702	29	0.581
2	0.9261	16	0.7583	30	0.419
3	1	17	0.803	31	0.4418
4	0.9785	18	0.7552	32	0.5358
5	0.8973	19	0.8071	33	0.4084
6	0.936	20	0.7806	34	0.5614
7	0.9654	21	0.7275	35	0.6097
8	0.9227	22	0.5593	36	0.3825
9	0.9589	23	0.4696	37	0.465
10	0.9815	24	0.5929	38	0.4149
11	0.7649	25	0.3947	39	0.3738
12	0.7359	26	0.4873	40	0.3811
13	0.8001	27	0.6321	41	0.4092
14	0.7961	28	0.5724		

Setelah diperoleh nilai *similarity* kasus uji dengan semua kasus yang memiliki indeks yang sama, maka langkah selanjutnya adalah mencari nilai *similarity* terbesar. Berdasarkan nilai *similarity* yang diperoleh pada tabel 3, dapat diketahui bahwa nilai *similarity* terbesar adalah 1 dan terdapat pada kasus ke-3 dimana pada tabel kasus dapat diketahui bahwa kasus 3 merupakan kasus yang memiliki id_penyakit 1. Kemudian id_penyakit yang diperoleh digunakan untuk mengambil nama penyakit pada tabel penyakit sehingga dapat diambil keputusan bahwa kasus uji tersebut memiliki kemiripan yang paling tinggi terhadap kasus 1 yang memiliki id_penyakit 1 yang merupakan penyakit Otitis Media Akut.

Proses pengujian tersebut dilakukan terhadap 111 kasus baru. Kasus baru yang digunakan pada pengujian merupakan kasus yang pernah terjadi sebelumnya, namun tidak digunakan pada basis kasus. Tabel 6.3 menunjukkan nilai *similarity* kasus-kasus yang diuji terhadap sistem. Hasil uji coba menunjukkan, dari 111 kasus uji, terdapat 9 kasus yang memiliki nilai *similarity* di bawah 0.8 dan 102 kasus yang memiliki nilai *similarity* di atas 0.8. Gambar 11 merupakan grafik hasil pengujian terhadap 111 data kasus uji.



Gambar 11 Grafik hasil pengujian 111 kasus baru.

4. KESIMPULAN

1. Penggunaan metode backpropagation pada proses *indexing* dapat membantu sistem dalam melakukan *retrieval* karena dengan menggunakan backpropagation, pencarian nilai *similarity* cukup dilakukan terhadap kasus yang memiliki indeks yang sama dengan kasus baru. Namun dalam proses pelatihan, *backpropagation* memerlukan waktu yang cukup lama karena harus mencoba parameter pelatihan satu per satu untuk memperoleh jaringan yang terbaik.
2. Sistem akan memberikan solusi apabila kasus baru memiliki nilai *similarity* lebih besar dari 0.8 terhadap kasus lama.
3. Hasil uji coba dari 111 kasus uji, terdapat 9 kasus yang memiliki nilai *similarity* di bawah 0.8.
4. Ada 2 kondisi revisi kasus: pertama, kasus yang didiagnosa tidak mempunyai kemiripan sama sekali dengan kasus-kasus yang ada dalam *case-base*. Kedua, kasus memiliki kemiripan dengan kasus yang ada dalam *case-base* tetapi memiliki nilai *similarity* dibawah 0.8.

5. SARAN

1. Perlu dilakukan penelitian terhadap metode *indexing* selain *backpropagation* sehingga dapat diperoleh nilai perbandingan hasil *indexing* dengan metode-metode lain.
2. Perlu dilakukan penelitian untuk domain penyakit yang berbeda, sehingga diharapkan dapat dihasilkan sebuah sistem terpadu yang dapat mendiagnosa penyakit dari berbagai domain, khususnya untuk domain penyakit yang umum terjadi pada masyarakat.

DAFTAR PUSTAKA

- [1]Salton, G. dan McGill, M. J., 1983, *Introduction to Modern Information Retrieval*, McGraw-Hill.
- [2]Salem, A.M., Roushdy, M., Hodhod, R.A., 2004, A Case based expert system for supporting diagnosis of heart diseases. *The International Journal of Artificial Intelligence and Machine Learning*, Vol.05.
- [3]Kim Kyung-Sup dan Ingoo Han, 2001, The cluster-*Indexing* method for case-based reasoning using self-organizing maps and learning vector quantization for bond rating cases, *Expert Systems with Applications*, Vol.21, pp 147-156.
- [4]Farhan M., Manaf M., Norwawi N., Wahab M., 2011, Faster Case Retrieval Using Hash *Indexing* Technique, *International Journal of Artificial Intelligence and Expert Systems (IJAE)*, Vol. (2): Issue (2).
- [5]Kusumadewi S., 2004, *Membangun Jaringan Syaraf Tiruan (Menggunakan Matlab & Excel Link)*, Graha Ilmu, Yogyakarta.
- [6]Zhu S., Wu J., Hui X., dan Guoping X., 2011, Scaling Up Top-K Cosine Similarity Search, *Data & Knowledge Engineering*, Vol. 70, pp 60-83.