

IMPLEMENTASI *CASE BASED REASONING* UNTUK SISTEM DIAGNOSIS PENYAKIT ANJING

Fransica Octaviani S.⁽¹⁾
foctas@yahoo.com

Joko Purwadi⁽²⁾
jokop@ukdw.ac.id

Rosa Delima⁽³⁾
rosa@ukdw.ac.id

Abstraksi

Penalaran Berbasis Kasus / Case Based Reasoning (CBR) merupakan metode yang dipergunakan untuk membangun sebuah sistem berbasis pengetahuan. Sumber pengetahuan sistem diperoleh dengan mengumpulkan penanganan kasus-kasus oleh seorang ahli/pakar. Keuntungan dengan penerapan metode ini adalah membangun pengetahuan tidak perlu melakukan akuisisi pengetahuan secara langsung dengan seorang pakar. Pada penelitian ini penulis mengimplementasikan metode CBR untuk membantu pendiagnosian penyakit anjing. Kasus-kasus yang dipergunakan dalam sistem diperoleh dari catatan penanganan kasus diagnosa penyakit anjing dari seorang dokter hewan. Sistem akan memberikan keluaran berupa kemungkinan penyakit dan saran pengobatan yang didasarkan pada kemiripan kasus baru dengan pengetahuan yang dimiliki sistem.

Kata Kunci : *Case Based Reasoning, Sistem Berbasis Pengetahuan, Knowledge Based System, penyakit anjing.*

1. Pendahuluan

1.1. Latar Belakang

Sebuah sistem berbasis pengetahuan memiliki dua elemen utama yaitu basis pengetahuan/*knowledge based* dan kemampuan penalaran/*reasoning*. Basis pengetahuan merupakan elemen utama sistem karena komponen ini berisi sumber kecerdasan sistem. Banyak metode yang dapat digunakan untuk membangun sebuah basis pengetahuan diantaranya melalui interaksi langsung pembangun pengetahuan dengan ahli/pakar melalui wawancara dan observasi atau melalui catatan penanganan kasus yang pernah dilakukan oleh seorang ahli. Akuisisi pengetahuan melalui catatan penanganan kasus yang pernah dilakukan oleh ahli memiliki banyak keuntungan diantara pengembang sistem tidak perlu berhubungan langsung dengan pakar dan proses akuisisi dapat lebih singkat, sehingga memperpendek waktu pengembangan sistem. Metode yang digunakan untuk membangun sebuah sistem berbasis pengetahuan yang

¹ Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Duta Wacana

² Teknik Informatika, Fakultas Teknologi Informasi Univeristas Kristen Duta Wacana

³ Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Duta Wacana

pengetahuannya bersumber dari catatan kasus-kasus lampau dikenal dengan *case based reasoning*(CBR).

Pada penelitian ini penulis menerapkan metode CBR untuk membangun sebuah sistem yang memiliki kemampuan mendiagnosa penyakit pada anjing. Proses diagnosa penyakit pada anjing memang sebaiknya dilakukan oleh seorang pakar yang merupakan seorang dokter hewan, namun dikarenakan anjing merupakan hewan peliharaan yang umum dimiliki oleh masyarakat dan biaya untuk konsultasi ke dokter hewan juga cukup mahal maka sebagian masyarakat yang memiliki anjing biasanya melakukan pengobatan sendiri terhadap anjing mereka yang sedang sakit. Minimnya pengetahuan yang dimiliki oleh masyarakat dapat menyebabkan penanganan yang salah terhadap anjing peliharaan mereka dan hal ini dapat berdampak fatal dan tidak jarang mengakibatkan kematian pada hewan peliharaannya. Keberadaan sistem ini diharapkan dapat membantu masyarakat dalam melakukan diagnosa awal terhadap penyakit yang diderita anjing mereka dan memberikan saran pengobatan terhadap penyakit tersebut.

2. Landasan Teori

2.1. Case-Based Reasoning

Case-Based Reasoning (CBR) adalah metode untuk menyelesaikan masalah dengan mengingat kejadian-kejadian yang sama/sejenis (*similar*) yang pernah terjadi di masa lalu kemudian menggunakan pengetahuan/informasi tersebut untuk menyelesaikan masalah yang baru, atau dengan kata lain menyelesaikan masalah dengan mengadaptasi solusi-solusi yang pernah digunakan di masa lalu.

Menurut Aamodt dan Plaza (1994) *Case-Based Reasoning* adalah suatu pendekatan untuk menyelesaikan suatu permasalahan (*problem solving*) berdasarkan solusi dari permasalahan sebelumnya. *Case-based Reasoning* ini merupakan suatu paradigma pemecahan masalah yang banyak mendapat pengakuan yang pada dasarnya berbeda dari pendekatan utama AI lainnya. Suatu masalah baru dipecahkan dengan menemukan kasus yang serupa di masa lampau, dan menggunakannya kembali pada situasi masalah yang baru. Perbedaan lain dari CBR yang tidak kalah penting adalah CBR juga merupakan suatu pendekatan ke arah *incremental* yaitu pembelajaran yang terus-menerus.

Dalam *Case-Based Reasoning* ada empat tahapan yang meliputi :

a. Retrieve

Mendapatkan/memperoleh kembali kasus yang paling menyerupai/relevan (*similar*) dengan kasus yang baru. Tahap *retrieval* ini dimulai dengan menggambarkan/menguraikan sebagian masalah, dan diakhiri jika ditemukannya kecocokan terhadap

masalah sebelumnya yang tingkat kecocokannya paling tinggi. Bagian ini mengacu pada segi identifikasi, kecocokan awal, pencarian dan pemilihan serta eksekusi.

b. **Reuse**

Memodelkan/menggunakan kembali pengetahuan dan informasi kasus lama berdasarkan bobot kemiripan yang paling relevan ke dalam kasus yang baru, sehingga menghasilkan usulan solusi dimana mungkin diperlukan suatu adaptasi dengan masalah yang baru tersebut.

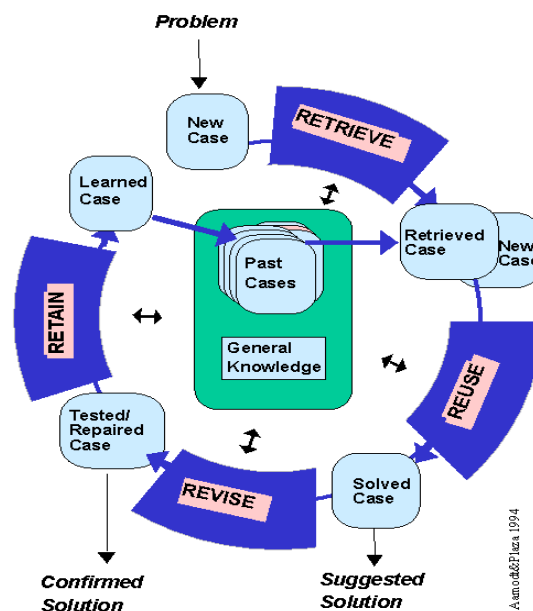
c. **Revise**

Meninjau kembali solusi yang diusulkan kemudian mengetesnya pada kasus nyata (simulasi) dan jika diperlukan memperbaiki solusi tersebut agar cocok dengan kasus yang baru.

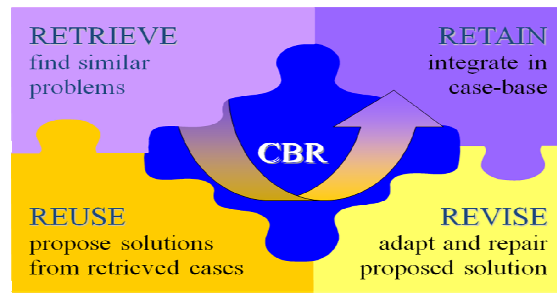
d. **Retain**

Mengintegrasikan/menyimpan kasus baru yang telah berhasil mendapatkan solusi agar dapat digunakan oleh kasus-kasus selanjutnya yang mirip dengan kasus tersebut. Tetapi Jika solusi baru tersebut gagal, maka menjelaskan kegagalannya, memperbaiki solusi yang digunakan, dan mengujinya lagi.

Empat proses masing-masing melibatkan sejumlah langkah-langkah spesifik, yang akan dijelaskan pada gambar 1 dan gambar 2 berikut ini



Gambar 1. Siklus Metode Case-Based Reasoning (Aamodt & Plaza, 1994)



Gambar 2. R⁴ Cycle dari *Case-Based Reasoning*

Pada saat terjadi permasalahan baru, pertama-tama sistem akan melakukan proses *retrieve*. Proses *retrieve* akan melakukan dua langkah pemrosesan, yaitu pengenalan masalah dan pencarian persamaan masalah pada database.

Setelah proses *retrieve* selesai dilakukan, selanjutnya sistem akan melakukan proses *reuse*. Di dalam proses *reuse*, sistem akan menggunakan informasi permasalahan sebelumnya yang memiliki kesamaan untuk menyelesaikan permasalahan yang baru. Pada proses *reuse* akan menyalin, menyeleksi, dan melengkapi informasi yang akan digunakan.

Selanjutnya pada proses *revise*, informasi tersebut akan dikalkulasi, dievaluasi, dan diperbaiki kembali untuk mengatasi kesalahan-kesalahan yang terjadi pada permasalahan baru.

Pada proses terakhir, sistem akan melakukan proses *retain*. Proses *retain* akan mengindeks, mengintegrasikan, dan mengekstrak solusi yang baru tersebut kedalam database. Selanjutnya, solusi baru itu akan disimpan ke dalam basis pengetahuan (*knowledge-base*) untuk menyelesaikan permasalahan yang akan datang. Tentunya, permasalahan yang akan diselesaikan adalah permasalahan yang memiliki kesamaan dengannya.

2.2 Penerapan Algoritma *Nearest Neighbor Retrieval* Pada Metode *Case-Based Reasoning*

Algoritma *Nearest Neighbor Retrieval* (*k-nearest neighbor* atau *k-NN*) adalah sebuah algoritma untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Kasus khusus di mana klasifikasi diprediksikan berdasarkan data pembelajaran yang paling dekat (dengan kata lain, $k = 1$) disebut algoritma *nearest neighbor*.

Algoritma *nearest neighbor* berdasarkan pada proses pembelajaran menggunakan analogi / *learning by analogi*. *Training* sampelnya dideskripsikan dalam bentuk atribut numerik *n*-dimensi. Tiap sampel mewakili sebuah titik pada ruang *n*-dimensi. Dengan cara ini, semua *training* sampel disimpan pada pola ruang *n*-dimensi. Ketika diberikan “*unknown*” sampel, *k-nearest neighbor classifier* mencari pola ruang *K training* sampel

yang paling dekat “*unknown*” sampel tersebut. *K training* sampel ini adalah *k nearest neighbor* dari *unknown* sampel. *Unknown* sampel ditetapkan dengan class yang paling umum diantara *k nearest neighbors*nya. Ketika $k = 1$, *unknown* sampel ditetapkan dengan class dari *training* sampel yang paling dekat dengan pola ruangnya.

Algoritma *nearest neighbor retrieval* menyimpan semua *training* sampel dan tidak membangun *classifier* sampai sampel baru (*unlabeled*) perlu diklasifikasikan, sehingga algoritma *nearest neighbor retrieval* sering disebut dengan *instance-based* atau *lazy learners*.

Rumus untuk menghitung bobot kemiripan (*similarity*) dengan *nearest neighbor retrieval* adalah:

$$\text{Similarity (problem,case)} = \frac{s_1 * w_1 + s_2 * w_2 + \dots + s_n * w_n}{w_1 + w_2 + \dots + w_n} \quad [2.1]$$

Keterangan:

S = *similarity* (nilai kemiripan)

W = *weight* (bobot yang diberikan)

3. Analisis Sistem

Metode *cased based reasoning* merupakan metode yang menerapkan 4 tahapan proses, yaitu *retrieve*, *reuse*, *revise*, dan *retain*. Cara kerja sistem secara umum berpedoman pada basis pengetahuan yang dimiliki oleh sistem yang bersumber dari kasus-kasus yang pernah ditangani oleh seorang dokter hewan (gambar 3.) yang kemudian dihitung tingkat kemiripannya dengan kasus baru yang dimasukan pengguna (gambar 4). Berdasarkan tingkat kemiripan kasus inilah sistem akan mengeluarkan diagnosis penyakit yang diderita oleh anjing berikut saran pengobatannya.

Kasus Lama : ID 81	Kasus Lama : ID 137	Kasus Lama : ID 74
<p>Gejala anjing Goby: Diare darah Anemia Dehidrasi Lemas Tidak mau makan Diagnosa: Penyakit: Coccidiosis</p>	<p>Gejala anjing Doggy: Diare darah Anemia Dehidrasi Lemas Diagnosa: Penyakit: Coccidiosis</p>	<p>Gejala anjing Pokky: Muntah kuning Diare darah Dehidrasi Tidak mau makan Anemia Diagnosa: Penyakit: Parvovirus (CPV)</p>
<p>Solusi: Sulfadimethoxine 55 mg/kg PO untuk hari pertama, kemudian 27,5 mg/kg/hari PO selama 4 hari</p>	<p>Solusi: Sulfadimethoxine 55 mg/kg PO untuk hari pertama, kemudian 27,5 mg/kg/hari PO selama 4 hari</p>	<p>Solusi: Ciprofloxacin dosis 10-15 mg/kg, 2x sehari. Boleh ditambah vitamin B-kompleks 1-5ml/kg, 2x sehari Dan infus ringer laktat intravena 20ml/kg per hari</p>

Gambar 3. Contoh kasus-kasus penyakit anjing

Kasus Baru X
Gejala anjing x: Diare darah Dehidrasi Tidak mau makan Anemia

Gambar 4. Contoh kasus baru.

a. Proses Retrieve

Proses *Retrieve* merupakan proses pencarian kemiripan kasus baru dengan kasus yang lama. Pencarian kemiripan antara kasus baru dengan kasus lama dilakukan dengan cara mencocokkan gejala yang diinputkan oleh pengguna dengan gejala yang ada pada basis pengetahuan. Pada proses *Retrieve* ini akan dilakukan pembobotan dengan menggunakan metode *Nearest Neighbour Retrieval*.

Pada awal proses diagnosa pengguna akan menginputkan gejala-gejala yang dialaminya. Pengguna dapat menekan tombol ‘lanjut’ terus-menerus sampai pertanyaan parameter gejala yang terakhir atau jika pengguna merasa bahwa gejala yang dimasukkan sudah cukup maka pengguna dapat menekan tombol ‘proses’ untuk dapat langsung mengetahui hasil diagnosanya. Kemudian sistem akan melakukan proses pembobotan dengan melakukan pencocokan satu per satu antara gejala yang dimasukkan dengan data yang ada di dalam basis pengetahuan. Proses pembobotan yang dilakukan oleh sistem ditampilkan dalam perhitungan di bawah ini.

Bobot parameter (w) :

- Gejala Penting = 5
- Gejala Sedang = 3
- Gejala Biasa = 1

$$Similarity (problem, case) = \frac{S_1 * W_1 + S_2 * W_2 + \dots + S_n * W_n}{W_1 + W_2 + \dots + W_n} \quad [2.1]$$

Keterangan:

S = *similarity* (nilai kemiripan) yaitu 1 (sama) dan 0 (beda)

W = *weight* (bobot yang diberikan)

Perhitungan kasus 1 :

Bobot gejala penyakit Coccidiosis :

- Diare darah = 5
- Anemia = 5

Dehidrasi = 3
 Lemas = 1
 Tidak mau makan = 1

Kasus Baru X		Kasus Lama : ID 81
Gejala anjing x:		Gejala anjing Goby:
Diare darah	5	Diare darah
Dehidrasi	5	Anemia
Tidak mau makan	3	Dehidrasi
Anemia	1	Lemas
		Tidak mau makan

$$\text{Similarity (X,81)} = \frac{[(1*5) + (1*5) + (1*3) + (0*1) + (1*1)]}{5 + 5 + 3 + 1 + 1}$$

$$= 0,9333$$

Perhitungan kasus 2 :

Kasus Baru X		Kasus Lama : ID 137
Gejala anjing x:		Gejala anjing Doggy:
Diare darah	5	Diare darah
Dehidrasi	5	Anemia
Tidak mau makan	3	Dehidrasi
Anemia		Lemas

$$\text{Similarity (X,137)} = \frac{[(1*5) + (1*5) + (1*3) + (0*1)]}{5 + 5 + 3 + 1}$$

$$= 0,9286$$

Bobot gejala penyakit Parvovirus (CPV)

Muntah kuning = 5
 Diare darah = 5
 Dehidrasi = 3
 Anemia = 3
 Tidak mau makan = 1

Perhitungan kasus 3 :

Kasus Baru X		Kasus Lama : ID 74
Gejala anjing x:		Gejala anjing Pokky:
Diare darah	5	Muntah kuning
Dehidrasi	3	Diare darah
Tidak mau makan	1	Dehidrasi
Anemia	3	Tidak mau makan
		Anemia

$$\text{Similarity (X,74)} = \frac{[(0*5) + (1*5) + (1*3) + (1*1) + (1*3)]}{5 + 5 + 3 + 3 + 1}$$

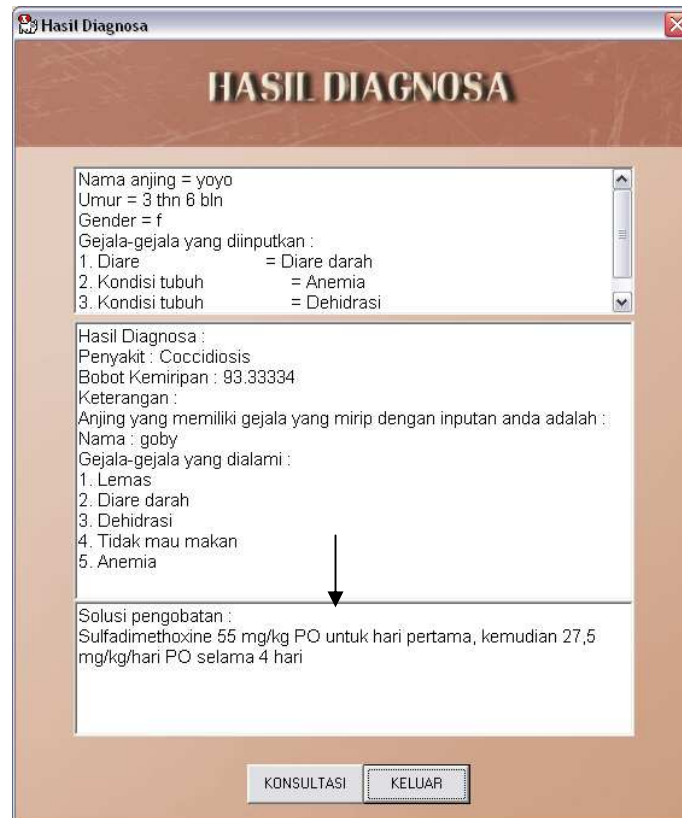
$$= 0,7058$$

b. Proses *Reuse*

Dari perhitungan diatas kasus yang memiliki bobot kemiripan paling rendah adalah kasus ID 74 yaitu sebesar 0,7058. Kasus ID 81 dan ID 137 menghasilkan bobot kemiripan yang hampir sama tinggi yaitu 0,9333 dan 0,9286, hanya selisih sedikit saja yaitu sebesar 0,0047.

Pada proses *reuse*, solusi yang diberikan adalah solusi dengan bobot kemiripan kasus lama dengan kasus baru yang paling tinggi, dalam contoh kasus ini adalah kasus ID 81 yaitu anjing Goby. Hasil perhitungan dengan bobot menunjukkan tingkat kepercayaan lebih dari 90%, jadi solusi kasus ID 81 lah yang direkomendasikan kepada pengguna yaitu Sulfadimethoxine 55 mg/kg PO untuk hari pertama, kemudian 27,5 mg/kg/hari PO selama 4 hari.

Pada gambar 5 dapat dilihat bahwa hasil perhitungan manual dan program cocok, yaitu menghasilkan *output* penyakit Coccidiosis beserta solusi pengobatannya.



Gambar 5. Gambar Penelusuran Penyakit Anjing

c. Proses *Revise*

Proses *revise* adalah proses peninjauan kembali kasus dan solusi yang diberikan jika pada proses *retrieve* sistem tidak dapat memberikan hasil diagnosa yang tepat. Pada contoh ini kasus anjing Goby sudah menghasilkan solusi dengan tingkat kepercayaan diatas 90%, jadi solusi yang dihasilkan dapat langsung diberikan.

Tetapi jika ternyata setelah dilakukan proses perhitungan dan tidak ada kasus yang mirip dengan kasus baru tersebut maka dilakukan proses *revise*. Informasi berupa masukan gejala pada kasus baru yang tidak ditemukan kemiripannya dengan basis pengetahuan (*rule*) tersebut akan ditampung pada suatu tabel khusus (tabel *revise*) yang selanjutnya akan dievaluasi dan diperbaiki kembali oleh pakar untuk menemukan solusi yang tepat.

d. Proses *Retain*

Setelah proses *revise* selesai dan sudah ditemukan solusi yang benar-benar tepat barulah pakar mulai menambah aturan dengan memasukkan data kasus baru yang sudah ditemukan solusinya tersebut ke dalam basis pengetahuan yang nantinya dapat digunakan untuk kasus berikutnya yang memiliki permasalahan yang sama. Proses inilah yang disebut dengan proses *retain*.

4. Kesimpulan

Berdasarkan hasil implementasi dan analisis sistem maka dapat disimpulkan :

- 1) Berdasarkan uji tes beta yang dilakukan, sistem pakar dapat melakukan diagnosis penyakit anjing dengan cukup baik.
- 2) Fasilitas modifikasi aturan (*rule*) yang diberikan memberikan kemudahan bagi pakar untuk mengubah aturan sehingga sistem ini dapat selalu diperbaharui dan dikembangkan sesuai kebutuhan.

Daftar Pustaka

- Aamodt, A & Plaza, E, 1994, Case Based Reasoning: Foundation Issues Methodological Variations, and System Approaches, AI Communication Vol 7 Nr, 1 March 1994, pp 39-59.
- Subronto, 2006. *Penyakit Infeksi Parasit dan Mikroba pada Anjing dan Kucing*. Yogyakarta: GADJAH MADA UNIVERSITY PRESS.
- Turban, E., Aronson, J.E., & Liang, T.P., 2005, *Decision Support System and Intelligent System*, New Jersey: Pearson Education, Inc.
- Watson, I., 1997, *Applying Case Based Reasoning: Techniques for Enterprise Systems*. San Fransisco: Morgan Kaufmann Publisher, Inc.