

PENERAPAN CASE-BASED REASONING UNTUK MENDIAGNOSIS PENYAKIT DIARE PADA MANUSIA

SANDY KOSASI

STMIK Pontianak, Jl. Merdeka No. 372 Pontianak
e-mail: sandykosasi@yahoo.co.id dan sandykosasi@stmikpontianak.ac.id

Abstract

Diarrhea disease is a very serious disease because it can bring death and usually attacks children at the age below five years. To minimize diarrhea disease, it needs fast and serious diagnosis to decrease the impact of death that tends to increase from time to time. The intelligent systems application design through case-based reasoning application can give simplicity to do an early diagnosis independently. This experiment is using reuse-based method with nearest-neighbor approach. Case-based reasoning has an ability to make diagnosis result more accurate according to the past incident and can be revised to solve the new problem. Result of the test gives a problem solution with similarity score 0,9. That means reach score 90%. Next the accuracy testing is using nearest-neighbor technique, between a new case and an early case that has the same case, we get the similarity result 1. It means that application has an accurate search. Test also shows consistent result because from some searches that have been done, it has the same similarity score

Keywords: case-based reasoning, diarrhea disease, similarity, reuse-based method, nearest-neighbor approach

1 PENDAHULUAN

Penyakit diare merupakan salah satu jenis penyakit yang paling banyak terjadi pada sistem saluran pencernaan manusia. Penyebab kematian karena penyakit diare tertinggi terjadi pada kelompok usia 29 hari - 11 bulan (31,4%), usia 1-4 tahun (25,2%), sisanya anak-anak muda dan orangtua yang sudah lanjut usia. Untuk penanganan penyakit diare ini sudah sering menjadi perhatian masyarakat umum, namun kenyataan yang ada memperlihatkan tidak semua masyarakat umum memahami dan mengetahui tindakan pencegahannya sehingga sampai saat ini masih tercatat jenis penyakit yang paling banyak menyebabkan kematian. Penelitian ini memiliki tujuan memberikan kemudahan untuk melakukan diagnosis secara awal mengenai masalah-masalah yang ditimbulkan oleh penyakit diare ini dengan menggunakan metode *Case-based reasoning*.

Metode *Case-Based reasoning* melakukan proses penyelesaian masalah dengan memanfaatkan pengalaman sebelumnya (Main et al. 2001). *Case-Based Reasoning* merupakan salah satu metode pemecahan masalah yang dalam mencari solusi dari suatu kasus yang baru, sistem akan melakukan pencarian terhadap solusi dari kasus lama yang memiliki permasalahan yang sama dan sudah pernah terjadi sebelumnya (Russel and Norvig 2014). Metode *Case-Based Reasoning* memiliki kelebihan dibandingkan dengan sistem berbasis aturan (*rule base system*) dalam hal akuisisi pengetahuan terletak pada kumpulan pengalaman/kasus-kasus sebelumnya. Metode ini tetap dapat melakukan penalaran walaupun terdapat data yang tidak lengkap. Ketika proses *retrieval* dilakukan,

ada kemungkinan antara kasus baru dengan kasus lama pada basis kasus tidak mirip. Namun, dari ukuran similarity tersebut tetap dapat dilakukan penalaran dan melakukan evaluasi terhadap ketidaklengkapan atau ketidaktepatan data yang diberikan (Pal and Shiu 2004). Perancangan *case-based reasoning* ini bisa digunakan untuk membantu pakar dalam mengidentifikasi penyakit dan memberi cara penanggulangannya. Hal ini tidak berarti menggantikan kedudukan pakar, tetapi hanya membantu dalam mengkonfirmasi keputusannya, karena mungkin bisa terdapat banyak alternatif yang harus dipilih secara tepat (Nugraheni 2008).

Penelitian sejenis yang pernah dilakukan cenderung hanya berbeda dari sisi pendekatan dalam melakukan penelusuran basis kasus dan metode untuk perhitungan similaritynya. Seperti halnya penelitian mengenai perancangan *case-based reasoning* menggunakan *sorenson coefficient* untuk perhitungan similaritynya (Nugraheni 2012). Penelitian mengenai teknologi penalaran berbasis kasus untuk mendiagnosa penyakit kehamilan menggunakan pendekatan *nearest neighbor* dalam perhitungan similaritynya (Muzid 2008). Penelitian pengembangan sistem cerdas menggunakan penalaran berbasis kasus untuk diagnosa penyakit akibat virus Eksantema menggunakan *probabilitas bayes* dalam perhitungan similaritynya (Aribowo 2010). Selanjutnya penelitian mengenai penalaran berbasis kasus untuk deteksi dini penyakit leukemia dengan memadukan metode *indexing* dan *Nearest Neighbor* dalam perhitungan similaritynya (Aribowo dan Khomsah 2012). Merujuk penelitian yang sudah dilakukan sebelumnya, dalam penelitian ini menggunakan pendekatan *reuse-based* dengan pendekatan *nearest neighbor* untuk perhitungan similaritynya.

2 METODE PENELITIAN

Penelitian ini menggunakan bentuk studi literatur dan metode penelitian eksperimen semu. Sementara untuk metode perancangan aplikasinya menggunakan *reuse-based* yang meliputi enam tahap yaitu spesifikasi persyaratan, analisis komponen, modifikasi persyaratan, integrasi design sistem dengan reuse, pengembangan dan integrasi, serta validasi sistem. Teknik pengumpulan data menggunakan wawancara, observasi, dan penyebaran angket daftar pertanyaan ke sejumlah responden dengan pendekatan *purposive sampling*. Tahapan *case-based reasoning* meliputi *case retrieve* (penelusuran) dimulai dengan tahapan mengenali masalah dan berakhir ketika kasus yang ingin dicari solusinya telah ditemukan serupa dengan kasus yang telah ada. Adapun beberapa tahapan yang terdapat dalam *retrieve* adalah: (1) identifikasi masalah, (2) memulai pencocokan, dan (3) menyeleksi. Tahapan paling penting adalah menentukan tingkat similarity (*similarity*) antar kasus. Melalui perhitungan similarity antar kasus, maka dapat dibuat daftar terurut dari kasus-kasus yang mirip (*similar case*). *Case Reuse* merupakan suatu kasus dalam konteks kasus baru dan terfokus pada dua aspek, yaitu Perbedaan antara kasus yang ada dengan kasus yang baru. Terdapat 2 (dua) cara yang bisa digunakan untuk meng-*reuse* kasus yang telah ada, yaitu menggunakan ulang solusi dari kasus yang telah ada (*transformatial reuse*)

dan menggunakan ulang metode kasus yang ada untuk membuat solusi (*derivational reuse*). Rumus *Similarity* dapat dilihat dinomor 1.

$$\text{Rumus Similarity} = \text{Similarity}(T, S) = \sum_{i=1}^n f(T_i, S_i) \times p_i \dots \dots \dots (1)$$

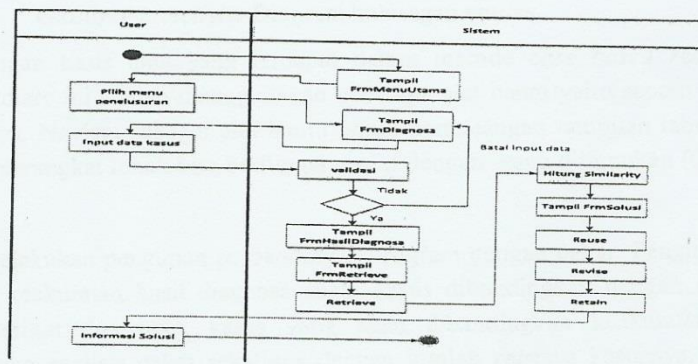
Keterangan :

T = Target
S = Kasus asal
w = Pembobotan atribut i
n = Jumlah atribut dalam setiap kasus
i = Atribut individu dari 1 ke n

3 HASIL DAN PEMBAHASAN

f = Nilai similarity nilai atribut i dalam kasus T dan S

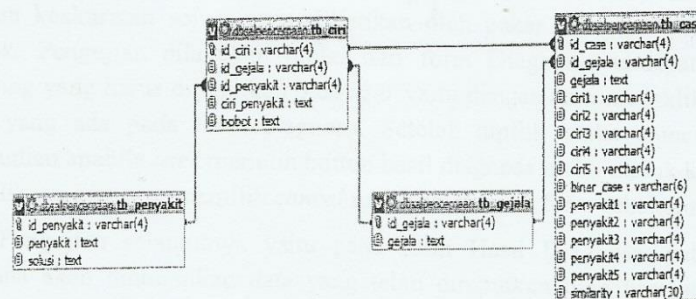
Untuk mengakses aplikasi yang disediakan oleh sistem, maka pengguna (*user*) dan admin harus melakukan login terlebih dahulu untuk melakukan diagnosa penyakit maka *user* harus memilih menu diagnosa setelah itu akan muncul form diagnosa yang berisi tentang gejala-gejala penyakit diare, dan *user* harus memilih salah satu gejala penyakit yang sudah tersedia. Setelah memilih gejala, maka *user* harus mencentang ciri-ciri penyakit yang sudah tersedia, berdasarkan ciri-ciri yang sudah dipilih oleh *user* maka sistem akan memprosesnya didalam database untuk mencari penyakit yang sesuai dengan ciri-ciri yang sudah dipilih, setelah itu akan muncul *output* berupa jenis penyakit dan solusi untuk penyakit yang sedang dialami. Admin bertugas untuk mengatur basis pengetahuan dan menambah data-data baru jika ditemukan jenis dan ciri-ciri gejala penyakit baru yang masih belum ada ke dalam basis pengetahuan (Gambar 1).



Gambar 1. Activity Diagram Penelusuran Kasus

Untuk mendapatkan solusi dari penyakit diare, *user* memilih menu *file* penelusuran kasus kemudian sistem menampilkan *FrmDiagnosa*. *User* melakukan input data kasus. Sistem akan mengecek validasi data tersebut. Jika invalid maka akan ditampilkan pesan kesalahan. Jika valid maka sistem menampilkan *frmHasilDiagnosa*. Kemudian sistem menampilkan *FrmRetrieve*, melakukan proses *retrieve* dan hitung similarity. Selanjutnya ditampilkan *FrmSolution*, solusi dari kasus dengan similarity terbesar akan digunakan kembali (*reuse*) sebagai solusi atau jika kasus baru maka akan direvisi (*revise*) dan diupdate dalam basis kasus (*retain*). Form login memiliki dua hak akses yaitu *user* dan *admin*, yaitu jika pengguna memilih *user*, maka tidak perlu lagi menginputkan *username*

dan *password* tetapi sebaliknya jika pengguna memilih *admin*, maka harus menginputkan *username* dan *password*. Setelah berhasil login, maka *user* dapat mengakses fitur-fitur yang ada di menu utama sesuai dengan fungsinya masing-masing. Jika *user* ingin melakukan diagnosa maka *user* bisa memilih menu diagnosa yang ada pada form utama dan pada saat form utama di jalankan, *user* akan di minta untuk memilih gejala penyakit yang di alami dan akan dilanjutkan dengan memilih ciri-ciri berdasarkan gejala yang sudah dipilih oleh *user*. Jika proses diagnosa selesai, sistem akan menampilkan *output* berupa jenis penyakit dan solusinya. Untuk melakukan pengolahan basis pengetahuan seperti *add*, *edit*, dan *delete*, maka *user* harus login sebagai *admin*.



Gambar 2. Activity Diagram hubungan entitas

Pada perancangan basis data yang terdapat dalam metode *case based reasoning* diagnosa penyakit diare ini adalah menggunakan beberapa alat bantu yaitu seperti MySQL Sebagai databasenya, Navicat sebagai alat bantu untuk perancangan tampilan tabel yang diperlukan supaya perangkat lunak bisa berfungsi sesuai dengan yang diinginkan (Gambar 2).

Selanjutnya melakukan pengujian perbandingan program dengan pakar. Pengujian ini untuk mengetahui keakuratan hasil diagnosa suatu kasus dibandingkan dengan analisis seorang pakar. Berikut dua buah kasus yang akan dibandingkan keakuratan nilai similaritasnya dengan analisis pakar sekaligus dengan jumlah *retrieve* kasusnya. Proses perbandingan disajikan dalam bentuk uraian berikut (tabel 1):

Kasus 1 :

Gejala : Saluran Pencernaan Atas (pada bagian Lambung dan Kerongkongan)
Ciri-ciri : Anda mengalami muntah, Anda mengeluh nyeri ulu hati, Anda merasa begah atau perut tidak nyaman karena terasa penuh, Anda muntah lagi setelah makan, Anda merasa perut lebih nyeri setelah makan

Tabel 1. Pengujian Perbandingan Program Dengan Pakar

Solusi kasus berdasarkan hasil perhitungan	Solusi kasus oleh pakar
--	-------------------------

Solusi dari penyakit ini Anda harus makan teratur dan jangan berlebihan, hindari alkohol, menghindari zat-zat pencetus maag, tidak meminum zat-zat berbahaya seperti asam dan basa buatan, meninggikan kepala saat tidur.

Solusi dari penyakit ini Anda harus makan teratur dan jangan berlebihan, sebisanya mengatur gaya hidup dan pola makan yang sehat, hindari rokok dan alkohol, rajin berolahraga, perbanyak makan buah dan sayur-sayuran hijau.

Jumlah *retrieve* : 5

Similarity : 0,9

Keakuratan : 90%

Hasil pengujian perbandingan keakuratan program dengan pakar pada tabel diatas dapat disimpulkan bahwa program yang dirancang mampu mendiagnosa penyakit dan memberikan solusi penyelesaian masalah yang dialami oleh *user* dengan nilai similaritas 0,9 dan keakuratan solusi yang diberikan oleh pakar sebesar 90%. Pengujian perangkat lunak. Pengujian dilakukan mulai dari form Diagnosa, terdapat beberapa pertanyaan penting yang harus diinputkan oleh *user* yaitu dengan cara mengklik tombol button tampil ciri yang ada pada form diagnosa. Setelah dipilih maka muncul kode ciri penyakit kemudian apabila *user* memilih button hasil diagnosa maka masuk ke form berikutnya tapi sebaliknya jika *user* memilih *cancel* maka kembali ke form sebelumnya.

Pengujian selanjutnya yaitu pada form Hasil Diagnosa, yaitu perancangan form dimana akan ditampilkan data yang telah diinputkan oleh *user* sebelumnya agar dapat mengetahui hasil yang telah diinputkan. Selanjutnya apabila *user* memilih button tampilan maka muncul informasi yang telah diinputkan sebelumnya kemudian jika memilih *retrieve* maka muncul form berikutnya.

Pengujian selanjutnya yaitu pada form Retrieve, yaitu form yang akan memberikan suatu perhitungan similarity atau suatu persamaan dari hasil nilai yang telah diinputkan sebelumnya. Perancangan form ini menampilkan jenis penyakit dan solusi dengan membandingkan atau menggunakan kembali solusi dari kasus-kasus yang lama dengan kasus-kasus yang ada dengan membandingkan nilai tertinggi. Pengujian selanjutnya yaitu pada form Reuse, yaitu form dimana akan memberikan solusi atau penjelasan tentang masalah yang telah diinputkan sebelumnya kepada *user*. Form solusi ini menampilkan jenis penyakit dan solusi dari kasus lama dengan memiliki nilai yang tertinggi. Kemudian jika *user* ingin mengakhiri program maka klik pada button keluar maka program yang dijalankan tadi telah selesai. Berikutnya melakukan pengujian basis kasus. Pengujian basis kasus meliputi dua proses, yaitu verifikasi dan validasi. Pengujian verifikasi berfokus memastikan sistem memberikan jawaban yang benar. Pengujian verifikasi yang dilakukan penulis terdiri dari:

3.1 Pengujian Keakuratan Penelusuran

Pengujian keakuratan penelusuran menggunakan teknik penelusuran *nearest-neighbor* yaitu dengan membandingkan kasus baru (kasus target) dengan kasus awal yang memiliki kasus yang sama. Kasus awal dalam basis kasus seharusnya tepat cocok pada dirinya sendiri sehingga tingkat *similarity*nya adalah 100%. (tabel 2 dan 3).

Tabel 2. Input Data Kasus Pengujian Keakuratan Penelusuran

Input Data Kasus	
Gejala	Saluran Pencernaan Atas (Muntah dan kekenyangan)
Ciri 1	Anda mengalami muntah.
Ciri 2	Anda merasa begah atau perut tidak nyaman karena terasa penuh.
Ciri 3	Anda mengeluh nyeri ulu hati.
Ciri 4	Diare sesudah mual atau muntah.
Ciri 5	Anda merasa perut lebih nyeri setelah makan.

Tabel 3. Kasus Awal Pengujian Keakuratan Penelusuran

Input Data Kasus	
Gejala	Saluran Pencernaan Atas (Muntah dan kekenyangan)
Ciri 1	Anda mengalami muntah.
Ciri 2	Anda merasa begah atau perut tidak nyaman karena terasa penuh.
Ciri 3	Anda mengeluh nyeri ulu hati.
Ciri 4	Diare sesudah mual atau muntah.
Ciri 5	Anda merasa perut lebih nyeri setelah makan.

$$\text{Similarity} = (2*1) + (4*1) + (6*1) + (8*1) + (10*1) = 1$$

$$\text{Tingkat similarity} = 1 * 100\% = 100\%$$

Berdasarkan pengujian keakuratan penelusuran, antara kasus baru dengan kasus awal yang memiliki kasus yang sama didapatkan hasil *similarity* 1. Berarti sistem yang dirancang sudah memiliki penelusuran yang akurat.

3.2 Pengujian Kekonsistenan Penelusuran

Pengujian kekonsistenan penelusuran adalah dengan memeriksa apakah kinerja penelusuran dua kali pencarian akan memberikan solusi yang sama. Penelusuran dilakukan pada kasus dengan sumber yang sama dengan keakuratan yang sama (tabel 4,5,dan 6).

Tabel 4. Input Data Kasus Uji Kekonsistenan Penelusuran

Input Data Kasus	ID
------------------	----

Gejala	Saluran Pencernaan Atas (Muntah dan kekenyangan)	gal
Ciri 1	Anda mengalami muntah.	1001
Ciri 2	Anda merasa begah atau perut tidak nyaman karena terasa penuh.	1002
Ciri 3	Anda mengeluh nyeri ulu hati.	1003
Ciri 4	Diare sesudah mual atau muntah.	1004
Ciri 5	Anda merasa perut lebih nyeri setelah makan.	1005

Tabel 5. Pengujian Kekonsistenan Penelusuran

ID Gejala	ID Ciri 1	ID Ciri 2	ID Ciri 3	ID Ciri 4	ID Ciri 5
gal	1001	1002	1003	1004	1005
gal	1001	1002	1003	1004	-
gal	1001	1002	1003	-	-
gal	1001	1002	-	-	-
gal	1001	-	-	-	-

Tabel 6. Pengujian Kekonsistenan Penelusuran (Lanjutan)

ID Case	ID Gejala	ID Ciri 1	ID Ciri 2	ID Ciri 3	ID Ciri 4	ID Ciri 5	Similarity
C01	gal	1001	1002	1003	1004	1005	0,94
C02	gal	1001	1002	1003	1004	-	0,87
C03	gal	1001	1002	1003	-	-	0,79
C04	gal	1001	1002	-	-	-	0,65
C05	gal	1001	-	-	-	-	0,52

Pengujian yang dilakukan terlihat konsisten, karena memiliki nilai *similarity* yang sama sehingga untuk beberapa kali penelusuran dilakukan maka tetap memberikan solusi yang sama.

4 KESIMPULAN

Berdasarkan hasil perancangan dan pengujian sistem, maka dapat ditarik beberapa kesimpulan, yaitu a) perangkat lunak *case based reasoning* untuk mendiagnosa penyakit diare ini bisa digunakan untuk mendiagnosa penyakit yang dialami oleh *user* (pengguna)

dan memberikan solusi untuk mencegah atau mengobati penyakit diare; b) perangkat lunak *case based reasoning* ini hanya mampu mendiagnosa penyakit dari gejala-gejala atau keluhan ringan yang dirasakan masyarakat, dan gejala-gejala yang terlihat, seperti muntah, nyeri perut, susah buang air besar, diare, dan penurunan berat badan. Adapun saran yang dapat penulis berikan mengenai perancangan perangkat lunak *case based reasoning* untuk mendiagnosa penyakit diare ini adalah : a) perangkat lunak *case based reasoning* untuk mendiagnosa penyakit diare ini sebaiknya juga diterapkan dalam bahasa pemrograman berbasis web, agar bisa digunakan secara online dan lebih bermanfaat; b) perangkat lunak *case based reasoning* untuk mendiagnosa penyakit diare ini pada perhitungan nilai similaritynya masih kurang akurat.

DAFTAR PUSTAKA

- [1] Aryani, Adriana Sari., Indarto., dan Abdiansyah. 2008. *Penalaran Komputer Berbasis Kasus (Case Based Reasoning – CBR)*, Edisi Kesatu. Ardana Media : Yogyakarta.
- [2] Aribowo, Agus Sasmito. (2010) Pengembangan Sistem Cerdas Menggunakan Penalaran Berbasis Kasus (Case-Based Reasoning) Untuk Diagnosa Penyakit Akibat Virus Eksantema, *Telematika*, Vol.7 No.1, Juli 2010, pp. 11-22.
- [3] Aribowo, Agus Sasmito dan Khomsah, Siti. (2012) Penalaran Berbasis Kasus Untuk Deteksi Dini Penyakit Leukemia, *Seminar Nasional Informatika 2012 (semnasIF 2012)*, ISSN: 1979-2328 UPN "Veteran" Yogyakarta, 30 Juni 2012, pp. C-168 s/d C174.
- [4] Main, J. Dillon, T.S. Shiu, S. (2001) *A Tutorial on Case-Based Reasoning: Soft Computing in Case-Based Reasoning (Eds)*, Sprenger-Verlag, London, pp. 1-28
- [5] Nugraheni, Murien. (2012) Rancangan Case-Based Reasoning Menggunakan Sorenson Coefficient, *Jurnal Informatika*, Vol.6 No.1, Januari 2012.
- [6] Pal, S. K. dan Shiu, S. C. K. (2004) *Foundation of Soft Case-Based Reasoning*, John Willey and Sons, Inc., Hoboken, New Jersey.
- [7] Russel, Stuart J., Norvig, Peter. (2014) *Artificial Intelligence a Modern Approach*. 3rd Edition, Pearson Education Limited.