



Sistem Penalaran Berbasis Kasus Untuk Menentukan Penerima Beasiswa

Muhammad Nurtanzis Sutoyo

Program Studi Sistem Informasi - FTI USN Kolaka
Jl. Pemuda No. 339 Kolaka Sulawesi Tenggara

Mr.iyes@yahoo.co.id

Abstract

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. Penelitian ini mencoba untuk membangun suatu sistem Penalaran Berbasis Kasus untuk menentukan beasiswa. Untuk menghitung kemiripan terlebih dahulu dilakukan proses indexing terhadap kasus lama. Hal ini dilakukan agar pada proses pencarian nilai similarity kasus baru terhadap basis kasus dapat lebih efisien karena cukup menghitung nilai similarity kasus baru terhadap data kasus yang memiliki indeks yang sama. Hasil uji coba sistem menunjukkan bahwa sistem penalaran berbasis kasus ini membantu dalam menentukan usulan beasiswa.

Keywords— *Case Based Reasoning, Nearest Neighbor, Beasiswa*

I. PENDAHULUAN

Dalam dunia pendidikan khususnya di Perguruan Tinggi banyak sekali beasiswa yang ditawarkan kepada para mahasiswa. Baik beasiswa yang berasal dari pemerintah maupun dari pihak swasta. Salah satu beasiswa yang ditawarkan pemerintah adalah beasiswa Peningkatan Prestasi Akademik (PPA) dan Bantuan Biaya Miskin (BBM). Untuk mendapatkan beasiswa tersebut maka harus sesuai dengan aturan yang telah ditetapkan oleh pemerintah. Adapun kriteria yang ditetapkan sesuai buku pedoman Beasiswa BBM dan PPA Tahun 2015, yaitu: indeks prestasi akademik (IPK), semester, pekerjaan orangtua, pendapatan orang tua, dan jumlah tanggungan.

Begitu banyaknya mahasiswa yang mengajukan permohonan untuk menerima beasiswa dan tidak mungkin semua mahasiswa dapat dikabulkan. Oleh karena itu perlu dibangun sebuah sistem dengan pendekatan yang dapat digunakan untuk menentukan penerima beasiswa, yaitu sistem penalaran berbasis kasus (*case based reasoning*). *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 [1]. 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 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 dengan kasus baru sistem CBR cukup menghitung nilai kemiripan terhadap kasus yang ada pada kelompok yang sama dengan kasus baru. 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 *Nearest Neighbor*. Metode ini merupakan metode yang digunakan untuk menghitung kedekatan antara kasus baru dengan kasus lama, yaitu berdasarkan pada pencocokan bobot dari sejumlah fitur yang ada [2].

Telah banyak penelitian mengenai menentukan penerima beasiswa. Putra dan Hardiyanti [3] melakukan penelitian dengan membahas tentang penentuan penerima beasiswa dengan menggunakan Fuzzy MADM. Metode Fuzzy MADM digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu. Kriteria yang digunakan adalah nilai IPK, penghasilan orangtua, jumlah tanggungan dan jumlah saudara kandung. Hasil penelitian menunjukkan bahwa bobot yang diberikan pada setiap kriteria mempengaruhi hasil akhir penentuan calon penerima beasiswa.

Begitu pula penelitian yang berfokus pada *Case-Based Reasoning* banyak sekali ditemui dengan berbagai macam metode. Rismawan dan Hartati [4] membahas tentang diagnosa penyakit THT. Proses *indexing* kasus pada penelitian tersebut dengan menggunakan *backpropagation*. Sedangkan proses perhitungan *similarity* dengan menggunakan *Cosine Coefficient*. Hasil dari penelitian bahwa dengan 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.

Toba dan Tanadi [5] membahas tentang pengembangan pemesanan kain. Perhitungan untuk mendapatkan kesamaan antar kasus dengan menggunakan metode *Weighted Euclidean Distance*. Hasil yang dicapai dari penerapan *Case Based Reasoning* pada pemesanan kain sesuai dengan yang diharapkan, yaitu sistem dapat melakukan proses *retrieve* yaitu proses pengambilan data-data yang memiliki kesamaan dengan permasalahan yang sedang dihadapi. Selanjutnya data tersebut akan dicocokkan dan dihitung kedekatannya sehingga dapat diketahui data yang memiliki kesamaan tertinggi. Data dengan kesamaan tertinggi tersebut akan diambil solusinya. Selanjutnya sistem akan menjalankan proses *reuse*.

A. Deskripsi Sistem

Proses pada sistem dimulai dengan melakukan pembentukan indeks untuk kasus-kasus yang ada, indeks pada kasus yang ada diperoleh. Setelah diperoleh indeks dari setiap kasus yang ada, proses selanjutnya yang terjadi pada sistem adalah menginisialisasi kriteria-kriteria mahasiswa yang dianggap sebagai kasus baru.

Setelah memperoleh indeks dari kasus baru, maka selanjutnya sistem melakukan perhitungan nilai *similarity* kasus baru terhadap kasus yang lama yang memiliki indeks yang sama. Proses perhitungan *similarity* menggunakan *Nearest Neighbor*. Nilai *similarity* berkisar antara 0 sampai dengan 1. Apabila *similarity* kasus baru dengan salah satu kasus yang ada pada basis kasus lebih mirip, berarti kasus baru tersebut akan menggunakan solusi yang sama dengan kasus lama yang ada pada basis kasus.

B. Representasi Kasus

Pada kasus ini tentang menentukan beasiswa ini, terdapat 3 macam solusi utama, yaitu beasiswa BBM, PPA, dan tidak dapat beasiswa. Adapun kriteria-kriteria yang digunakan dan bobot untuk tiap kriteria untuk menentukan beasiswa, yaitu:

- IPK (C1) = 0.80
- Semester (C2) = 0.75
- Pekerjaan orangtua (C3) = 0.70
- Pendapatan orangtua (C4) = 0.65
- Jumlah tanggungan (C5) = 0.60

Dengan kedekatan antara nilai-nilai kriteria disajikan seperti Tabel 1.

TABEL 1
KEDEKATAN NILAI KRITERIA

Kriteria	Jika Sama	Jika Berbeda
C1	1	0,10
C2	1	0,45
C3	1	0,50
C4	1	0,35
C5	1	0,15

C. 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 kasus ini metode yang digunakan untuk *indexing* adalah kode Program Studi. Kode Program Studi mengikuti kode yang telah ditetapkan oleh DIKTI. Contoh kode program studi seperti Tabel 2.

TABEL II
KODE PROGRAM STUDI

Kode	Nama Program Studi
57201	Pend Bahasa Indonesia
57202	Pend Bahasa Inggris
57203	Pend Matematika

Apabila kode program studi kasus baru dibagi dengan kode kasus lama hasil sama dengan 1, maka kasus lama dari Program Studi itulah yang akan menjadi solusi sesuai nilai kemiripan.

D. Retrieval dan Similarity

Teknik retrieval dan similarity yang digunakan dalam kasus 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.

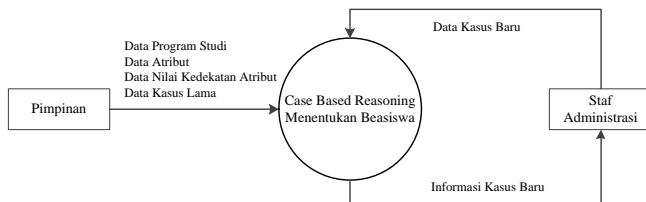
$$Similarity(T, S) = \frac{\sum_{i=1}^n f(T, S) * w_i}{\sum_{i=1}^n w_i}$$

Dimana T kasus baru, S kasus lama yang ada, n jumlah atribut, i atribut individu 1 s.d n , f kemiripan antara kasus T dan kasus S , dan w bobot yang diberikan.

E. Desain Sistem

Diagram Context

Gambar 1 merupakan diagram context yang menggambarkan sistem secara keseluruhan.



Gambar 1 Diagram Context

Diagram context terdiri dari 2 elemen lingkungan yaitu Staff Administrasi, dan Pimpinan serta sebuah proses secara keseluruhan dari sistem *case-based reasoning* untuk menentukan beasiswa. Pimpinan memberikan input ke sistem berupa data program studi, data atribut, nilai kedekatan atribut, dan kasus lama. Sedangkan staff administrasi akan memberikan *input* ke sistem berupa data kasus baru. Apabila kasus baru tidak ditemukan, maka proses revisi akan dilakukan oleh Pimpinan, dan hasil dari revisi akan disimpan sebagai basis kasus pada sistem.

F. Implementasi Sistem

Tahapan penggunaan sistem dimulai dengan melakukan pengambilan terhadap basis kasus terlebih dahulu, serta nilai

kedekatan atribut agar diperoleh nilai bobot yang nantinya akan dipergunakan pada saat ada kasus baru. Sistem akan melakukan *indexing* terhadap kasus baru dan menghitung kemiripan antara kasus baru dengan kasus lama. Gambar 2 merupakan tampilan sistem untuk halaman konsultasi. Pada halaman ini terdapat beberapa bagian yang berisi IPK, semester, pekerjaan orangtua, pendapatan serta jumlah tanggungan.

Gambar 2 Halaman Konsultasi

III. HASIL DAN PEMBAHASAN

Pada bagian ini akan dilakukan pengujian kasus terhadap sistem. Kasus baru yang akan diuji sebagai berikut.

- Program Studi = Pendidikan Matematika
- IPK = 3.20
- Semester = III
- Pekerjaan Orangtua = Wiraswasta
- Pendapatan Orangtua = 1.500.000
- Jumlah Tanggungan = 3 orang

Langkah selanjutnya adalah melakukan proses *indexing* terhadap kasus tersebut, yaitu proses menentukan kasus uji berada pada indeks/kelas yang mana. Hal ini dilakukan agar pada proses pencarian nilai *similarity* kasus baru terhadap basis kasus dapat lebih efisien karena cukup menghitung nilai *similarity* kasus baru terhadap data kasus yang memiliki indeks yang sama. Proses pencarian indeks diawali dengan membagikan nilai kode program studi kasus lama dan kode kasus baru. Jika hasil pembagian menghasilkan nilai 1, maka program studi tersebutlah yang akan dicari nilai kemiripan untuk kasus baru. Nilai hasil proses tersebut dapat dilihat pada Tabel 3.

TABEL 3
HASIL PROSES INDEXING

Kode Kasus Baru	Kode Kasus Lama	Hasil Diperoleh
57203	57201	0.999965
57203	57202	0.999983
57203	57203	1

Setelah memperoleh indeks dari kasus, langkah selanjutnya adalah menghitung nilai *similarity* dengan *Nearest Neighbor* dari kasus uji tersebut terhadap semua

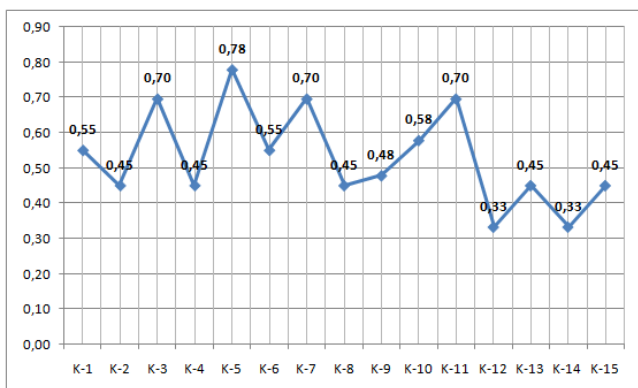
kasus pada basis kasus yang memiliki nilai indeks 1. Berdasarkan data kasus yang dimiliki, maka dapat diketahui bahwa kasus yang memiliki indeks 1 adalah program studi Pendidikan Matematika yang berjumlah 15 kasus.

Nilai *similarity* dengan *Nearest Neighbor* dihitung berdasarkan persamaan (1), dimana kasus dihitung nilai *similarity*-nya satu per satu terhadap 15 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 15 nilai. Tabel 4 menunjukkan nilai *similarity* yang diperoleh dari perhitungan *similarity* antara kasus baru dengan kasus ada yang pada basis kasus yang memiliki indeks yang sama.

TABEL 4
NILAI SIMILARITY KASUS

No	Kode	Kemiripan	Keterangan
1	K021	0,55	PPA
2	K022	0,45	PPA
3	K023	0,70	PPA
4	K024	0,45	PPA
5	K025	0,78	PPA
6	K026	0,55	BBM
7	K027	0,70	BBM
8	K028	0,45	BBM
9	K029	0,48	BBM
10	K030	0,58	BBM
11	K035	0,70	Tidak Dapat
12	K036	0,33	Tidak Dapat
13	K040	0,45	Tidak Dapat
14	K042	0,33	BBM
15	K043	0,45	BBM

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 4, dapat diketahui bahwa nilai *similarity* terbesar adalah 0.78 dan terdapat pada kasus ke-5 dimana pada tabel kasus dapat diketahui bahwa kasus ke-5 merupakan kasus yang memperoleh beasiswa PPA.



Gambar 3 Grafik Hasil Pengujian Kasus Baru

Gambar 3 di atas, merupakan grafik hasil pengujian terhadap kasus baru dengan kasus lama yang berjumlah 15 kasus.

IV. KESIMPULAN

Begitu banyak penggunaan metode pada proses *indexing* dapat membantu sistem dalam melakukan *retrieval* karena dengan menggunakan *indexing*, pencarian nilai *similarity* cukup dilakukan terhadap kasus yang memiliki indeks yang sama dengan kasus baru. Dari hasil pengujian kasus baru diperoleh nilai kemiripan tertinggi 0.78 dan terdapat pada kasus ke-5 dimana pada tabel kasus dapat diketahui bahwa kasus ke-5 merupakan kasus yang memperoleh beasiswa PPA. Hasil yang diperoleh dari kemiripan kasus baru terhadap kasus lama dapat dijadikan bahan pertimbangan (usulan) untuk mendapatkan beasiswa.

DAFTAR PUSTAKA

- [1] Aryani, AS., Indarto., Abdiansah., 2007, *Penalaran Komputer Berbasis Kasus (Case Based Reasoning)*, Ardana Media, Yogyakarta.
- [2] Kusri., dan Luthfi, ET., 2009, *Algoritma Data Mining*, Andri Offset, Yogyakarta.
- [3] Putra, Apriansyah dan Hardiyanti, DY., 2011, *Penentuan Penerima Beasiswa Dengan Menggunakan Fuzzy MADM*, Jurnal Sistem Informasi (JSI), Vol. 3, No. 1, April 2011.
- [4] Rismawan, Tedy dan Hartati, Sri., 2012, *Case-Based Reasoning Untuk Diagnosa Penyakit THT (Telinga Hidung dan Tenggorokan)*, IJCCS, Vol.6, No.2, July 2012, pp. 67~78.
- [5] Toba, Hapnes dan Tanadi, Sylvia., 2008, *Pengembangan Case-Based Reasoning Pada Aplikasi Pemesanan Kain Berdasarkan Studi Kasus Pada CV. Mitra KH Bandung*, Jurnal Informatika, Vol. 4, No.2, Desember 2008: 135 – 148.