

# Fungsi Similaritas Pada Sistem Berbasis Kasus Penyelesaian Masalah Akademik Mahasiswa

Syaiful Hendra<sup>1</sup>, Sri Kusumadewi<sup>2</sup>

Jurusan Teknik Informatika STMIK Adhi Guna  
Jl. Undata No. 3 Palu Sulawesi Tengah Indonesia  
Telp (0451) 455530, fax (0451) 455530  
syaiful.hendra.garuda@gmail.com<sup>1</sup>  
cicikusuma@yahoo.co.id<sup>2</sup>

**Abstract.** Penelitian ini mengusulkan salah satu pendekatan dalam identifikasi masalah akademik mahasiswa yaitu dengan pendekatan algoritma *K-Nearest Neighbor* (K-NN). Dalam sistem berbasis kasus penyelesaian masalah akademik mahasiswa yang dibangun, fungsi similaritas dari algoritma K-NN merupakan salah satu hal yang terpenting untuk menentukan nilai kemiripan antara kasus-kasus yang tersimpan di dalam basis kasus dengan kasus baru yang akan dicari solusinya. Hasil dari penelitian ini ditemukan bahwa fungsi similaritas pada algoritma K-NN dapat digunakan dalam menentukan nilai kedekatan antara kasus baru terhadap kasus lama.

**Keywords:** *Konseling Mahasiswa, Similaritas, K-Nearest Neighbor.*

## 1 Pendahuluan

### 1.1 Latar Belakang

Sistem berbasis kasus atau *case-based reasoning* (CBR) merupakan penalaran yang bertujuan untuk menyelesaikan permasalahan baru dengan cara mengadaptasi solusi-solusi yang terdapat pada kasus-kasus sebelumnya yang mempunyai permasalahan yang mirip dengan kasus yang baru<sup>1</sup>. Sumber pengetahuan utama sistem CBR adalah kasus-kasus yang telah ada atau yang telah tersimpan didalam basis kasus<sup>2</sup>. Kasus-kasus dapat diperoleh dari pengalaman seseorang atau pengalaman seorang pakar dibidangnya. Implementasi CBR sudah banyak dilakukan diberbagai bidang salah satunya di bidang konseling. Pada penelitian ini dilakukan identifikasi masalah akademik yang sering dialami mahasiswa pada perguruan tinggi dengan menggunakan metode *case-based reasoning* (CBR). Dalam pencarian solusi terbaik CBR menggunakan tahapan *retrieval* atau fungsi similaritas. Banyak teknik yang digunakan dalam menentukan nilai kemiripan atau kedekatan, salah satunya adalah teknik *nearest neighbor*. Algoritma K-NN sendiri merupakan pendekatan yang digunakan untuk menghitung nilai kemiripan atau kedekatan antar kasus.

Berdasarkan fenomena tersebut dirumuskan permasalahan dalam penelitian ini yaitu “Bagaimana menerapkan fungsi similaritas pada sistem berbasis kasus penyelesaian masalah akademik mahasiswa?”.

Sedangkan tujuan dari penelitian ini adalah untuk menerapkan metode dari fungsi similaritas yang ada pada algoritma *K-Nearest Neighbor* dalam mengidentifikasi masalah akademik mahasiswa dengan mencari tingkat kemiripan (*similarity*) antara basis kasus (kasus lama) dengan kasus yang akan diuji (kasus baru).

## 2 Landasan Teori

### 2.1 Sistem Pakar

Sistem pakar adalah program komputer yang menggunakan pengetahuan pakar untuk mencapai tingkat kinerja yang tinggi pada area yang sempit<sup>3</sup>. Secara umum, sistem pakar adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli<sup>4</sup>.

Kemudian Giarratano dan Riley<sup>5</sup> berpendapat bahwa sistem pakar adalah suatu sistem komputer yang bisa menyamai atau meniru kemampuan seorang pakar. Dari beberapa definisi tersebut dapat disimpulkan bahwa sistem pakar adalah sebuah sistem komputer yang diberi pengetahuan tentang kepakaran dalam bidang tertentu untuk membantu memberikan solusi terhadap masalah di bidangnya dengan meniru keahlian dari seorang pakar.

Sistem pakar dalam melakukan aktivitasnya menggunakan penalaran seperti yang dilakukan oleh seorang pakar, salah satu metode penalaran yang paling mirip dengan penalaran pakar adalah penalaran berbasis kasus atau *Case Based Reasoning (CBR)*. CBR harus melakukan beberapa tahapan proses untuk menghasilkan solusi suatu masalah diantaranya adalah mencari tingkat kemiripan kasus baru dengan kasus yang tersimpan.

Tahapan proses pada CBR dalam mencari nilai kemiripan dan proses pembelajaran dibutuhkan empat (4) tahap, yaitu : *retrieve*, *reuse*, *revise* dan *retain*<sup>1,6</sup>. Serta dibutuhkan 2 langkah utama dalam menentukan solusi terhadap kasus yaitu membangun basis kasus dan menentukan fungsi kemiripan (*similarity*)<sup>7</sup>.

### 2.2 Fungsi Similaritas K-Nearest Neighbor

*Nearest Neighbor* adalah pendekatan untuk mencari kasus dengan menghitung kedekatan antara kasus baru dengan kasus lama, yaitu berdasarkan pada pencocokan bobot dari sejumlah fitur yang ada. Misalkan diinginkan untuk mencari solusi masalah akademik dari seorang mahasiswa baru dengan menggunakan solusi dari masalah akademik terhadap mahasiswa terdahulu. Untuk mencari kasus mahasiswa mana yang akan digunakan maka dihitung kedekatan kasus mahasiswa baru dengan semua kasus mahasiswa lama. Kasus pada mahasiswa lama dengan kedekatan terbesar-lah yang akan diambil solusinya untuk digunakan pada kasus mahasiswa yang baru.

Misalkan terdapat 2 mahasiswa dengan kasus lama yaitu A dan B. Ketika ada mahasiswa Baru, maka solusi yang akan diambil adalah solusi dari mahasiswa terdekat dari mahasiswa Baru. Seandainya  $d_1$  adalah kedekatan antara mahasiswa Baru dengan mahasiswa A, sedangkan  $d_2$  adalah kedekatan antara mahasiswa Baru dengan mahasiswa B. Karena  $d_2$  lebih dekat dari  $d_1$  maka solusi dari mahasiswa B-lah yang akan digunakan untuk memberikan solusi pada mahasiswa Baru. Adapun rumus untuk melakukan penghitungan kedekatan antara dua kasus adalah sebagai berikut

$$Sim(T, S_i) = \frac{\sum_{i=1}^n f(T, S_i) w_i}{\sum_{i=1}^n w_i} \dots\dots\dots(1)$$

Keterangan :

- T = Kasus baru (*target*)
- S = Kasus yang ada dalam penyimpanan (*source*)
- n = Jumlah total atribut
- i = Jumlah atribut dalam masing-masing kasus
- f = Fungsi similarity atribut *i* antara kasus T dan kasus S
- w<sub>i</sub> = Bobot yang diberikan pada atribut ke *i*

Kedekatan biasanya berada pada nilai antara 0 s.d. 1. Nilai 0 artinya kedua kasus mutlak tidak mirip, sebaliknya untuk nilai 1 kasus mirip dengan mutlak.

### 2.3 Representasi Kasus

CBR tergantung pada struktur dan isi dari koleksi kasus. Suatu kasus dapat diselesaikan dengan memanggil kasus sebelumnya yang sesuai atau cocok dengan kasus baru. Sebuah kasus dapat menjadi sebuah catatan dari kejadian secara khusus terdiri dari [1]:

- a) Permasalahan (*problem*) yang menjelaskan keadaan nyata ketika kasus terjadi.
- b) Solusi, keadaan diperoleh / solusi dari permasalahan.

Pada representasi kasus sistem yang dibangun, setiap kasus dibagi menjadi 4 bagian utama yaitu:

- 1. Kondisi awal mahasiswa (keadaan psikologis, asal daerah)
- 2. Keluhan (ekonomi, keluarga, pribadi, lingkungan)
- 3. Masalah akademik.
- 4. Solusi

Pembagian ini dilakukan untuk memudahkan penyimpanan data kasus kedalam basis kasus, serta memudahkan dalam pengambilan data yang sesuai dengan kasus baru. Setiap kasus yang disimpan memiliki tiga bagian yang digunakan dalam memudahkan penyimpanan data kasus. Dari keempat bagian tersebut hanya dua bagian yaitu kondisi awal dan keluhan yang digunakan untuk dalam pencarian kasus yang mirip, sedangkan masalah akademik dan solusi merupakan solusi atau *output* dari sistem.

### 2.4 Proses Retrieval

Prose *retrieval* yang digunakan pada penelitian ini adalah dengan cara membandingkan atribut, antara atribut kasus baru dengan atribut kasus yang ada di dalam basis kasus, kemudian hasil perbandingan tersebut akan dihitung tingkat kemiripannya. Atribut yang digunakan adalah berupa kondisi awal dan keluhan yang terjadi pada mahasiswa. Perbandingan atribut menggunakan nilai bobot dinamis yang telah ditetapkan pakar konseling antara 1 - 5 semakin tinggi nilai bobot maka semakin penting sebuah atribut pada suatu kasus.

Perhitungan similaritas digunakan untuk menghasilkan nilai apakah ada kemiripan atau tidak antara kasus baru dengan kasus yang telah ada di basis kasus. Pada penelitian ini untuk menghitung similaritas antar kasus digunakan teknik *K-Nearest Neighbor*.

### 3 Representasi Kasus Masalah Akademik Mahasiswa

Penentuan atribut dari kasus yang akan dijadikan sebagai basis kasus. Pada penelitian ini hanya mengidentifikasi 10 masalah akademik mahasiswa menurut Kartini [8] dan fakta empiris dilapangan :

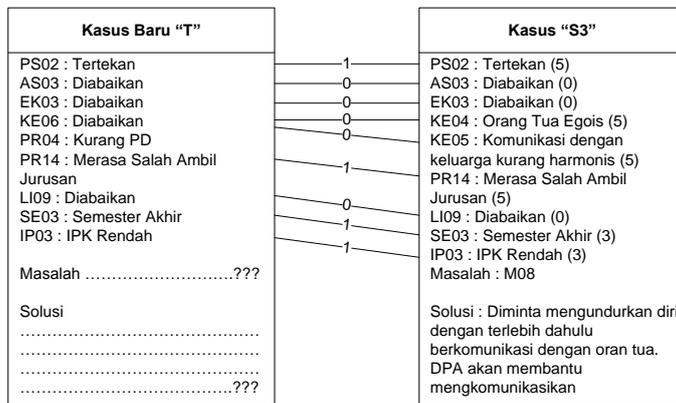
- M01.** Masalah dalam waktu belajar yang sesuai dengan banyak tuntutan dan aktivitas mahasiswa lainnya
- M02.** Masalah dalam memilih program yang sesuai dengan kemampuan.
- M03.** Masalah dalam menyusun tugas akhir/skripsi
- M04.** Masalah dalam mempelajari matakuliah tertentu
- M05.** Kurang motif atau semangat belajar
- M06.** Adanya kegiatan belajar yang salah
- M07.** Rendahnya rasa ingin tahu dan mendalami ilmu dalam rekayasa
- M08.** Kurangnya minat terhadap profesi.
- M09.** Tidak bisa / bingung melakukan perencanaan studi
- M10.** *Drop out / Terancam Drop out*

Tabel 1 merupakan atribut yang digunakan dalam penelitian ini. Berdasarkan hasil akusisi pengetahuan yang dilakukan dengan wawancara dari beberapa dosen pembimbing akademik.

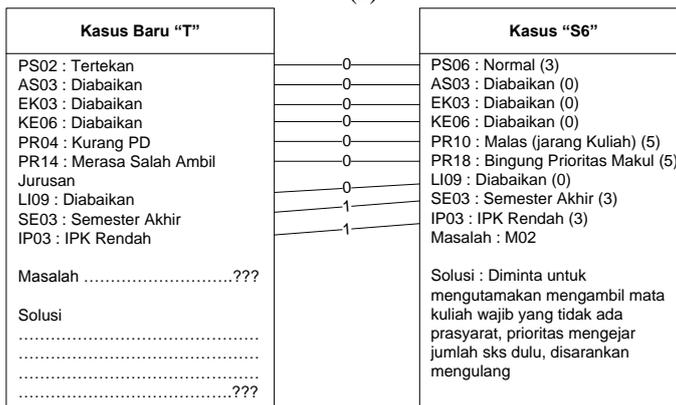
**Tabel 1. Atribut**

<b>Atribut</b>	<b>Kondisi</b>	<b>Sub Kondisi</b>	<b>Kode</b>
Keadaan Psikologis	a. Depresi	- Tertutup	PS01
		- Tertekan	PS02
		- Trauma masa lalu	PS03
		- Murung	PS04
	b. Diabaikan		PS05
	c. Normal		PS06
Asal	a. Asli daerah Universitas		AS01
	b. Luar kota Universitas		AS02
	c. Diabaikan		AS03
Ekonomi	a. Mampu		EK01
	b. Tidak mampu		EK02
	c. Diabaikan		EK03
Keluarga	a. Masalah	- Perceraian orang tua	KE01
		- Orang tua meninggal	KE02
		- Pertengkaran orang tua (KDRT)	KE03
		- Orang tua egois	KE04
		- Komunikasi dengan keluarga kurang harmonis	KE05
	b. Diabaikan		KE06
	c. Tidak Bermasalah		KE07
Lingkungan	a. Negatif	- Cara mengajar dosen membosankan	LI01
<b>Atribut</b>	<b>Kondisi</b>	<b>Sub Kondisi</b>	<b>Kode</b>
Lingkungan		- Tidak dapat akses data penelitian	LI02
		- Kompleksitas skripsi/tugas dari dosen	LI03
		- Dosen sekali masuk absensi lebih dari sekali	LI04
		- Dosen sulit ditemui	LI05
		- Dosen terkesan sulit memberi masukan	LI06
		- Salah paham / berselisih dengan dosen	LI07
		- Teman sering mengajak ke hal negatif (main game, narkoba, minum, judi dll)	LI08
		a. Diabaikan	
	b. Positif		LI10
	IPK	a. Tinggi	> 3,5
b. Sedang		2,75 – 3, 5	IP02
c. Rendah		< 2,75	IP03
d. Diabaikan			IP04
Semester	a. Awal	I sampai II	SE01
	b. Tengah	III sampai VI	SE02
	c. Akhir	VII sampai VIII	SE03
	d. Tua	IX sampai XVI	SE04
	e. Diabaikan		SE05
Pribadi	a. Tidak bermasalah	- Aktif dalam perkuliahan	PR01
		- Rajin	PR02
		- Cerdas	PR03
	b. Bermasalah	- Kurang percaya diri	PR04
		- Sibuk organisasi/kerja	PR05

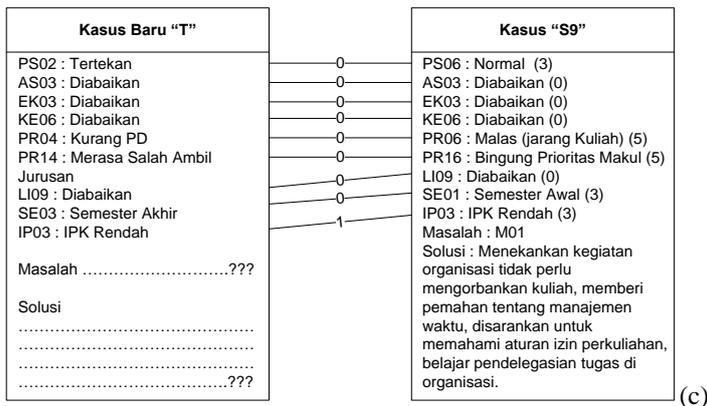
Contoh Kasus :



(a)



(b)



(c)

### Perhitungan Kemiripan

- (a) Tingkat kemiripan kasus baru/target (T) terhadap kasus S3  
 $Sim(S3, T) = (1*5)+(0)+(0)+(0)+(0)+(1*5)+(0)+(1*3)+(1*3)/(2*5)+(4*3)$   
 $= 16/22 = \mathbf{0.73}$
- (b) Tingkat kemiripan kasus baru (X) terhadap kasus S6  
 $Sim(S6, T) = (0)+(0)+(0)+(0)+(0)+(0)+(0)+(1*3)+(1*3)/(2*5)+(3*3)$   
 $= 6/19 = \mathbf{0.32}$
- (c) Tingkat kemiripan kasus baru (X) terhadap kasus S9  
 $Sim(S9, T) = (0)+(0)+(0)+(0)+(0)+(0)+(0)+(0)+(1*3)/(2*5)+(2*3)$   
 $= 3/16 = \mathbf{0.19}$

Berdasarkan hasil perhitungan kemiripan yang dilakukan pada contoh kasus, dapat diketahui bahwa kasus baru "T" memiliki tingkat kemiripan yang paling tinggi dengan kasus "S3" yaitu sebesar 0.73 atau 73%, sehingga jika yang sistem dibangun menggunakan nilai ambang batas 65% untuk *filtering* kasus, maka solusi yang ada pada kasus "S3" akan langsung direkomendasikan sebagai solusi atas kasus baru tersebut.

## 4 Keterbatasan dan Saran

Salah satu kelemahan yang terdapat pada fungsi similaritas ini yaitu masih memungkinkan untuk terjadinya "kemiripan palsu" dengan nilai 100%. Hal ini karena pada penelitian ini belum mempertimbangkan jumlah komponen yang terlibat dalam proses pencocokan antara kasus baru dengan kasus yang ada di dalam basis kasus. Jadi untuk penelitian mendatang akan dilakukan penelitian lebih lanjut serta perbaikan fungsi similaritas terkait dengan penentuan jumlah komponen yang terlibat dalam proses pencocokan.

Fungsi similaritas yang terdapat dalam penelitian ini masih dimungkinkan terdapat kemiripan palsu hal ini dikarenakan adanya peluang *user* untuk memilih atribut/komponen "diabaikan" yang memiliki nilai "0" sehingga jika ini diterapkan dalam basis kasus dimana pada saat terjadi pencocokan nilai yang terdapat pada kasus baru > dari kasus dalam basis kasus maka akan terjadi kemiripan palsu dengan nilai 100%. Berikut ilustrasi kemiripan palsu tersebut.

Misal

S1 : **PS06(5), AS03(0), EK02(3), KE06(0), PR01(5), PR02(5), LI09(0), SE03(3), IPK04(0).**

Total nilai bobot = 21

S2 : **PS06(5), AS03(0), EK02(3), KE06(0), PR25(0), LI09(0), SE03(3), IPK04(0)**

Total nilai bobot = 11

T : **PS06, AS03, EK02, KE06, PR01, PR02, LI09, SE03, IPK04.**

Similaritas

(T,S1) :  $5 + 3 + 5 + 5 + 3 = 21/21 = 1 / (100\%)$

(T,S2) :  $5 + 3 + 3 = 11/11 = 1 / (100\%) \leftarrow \text{Palsu}$

Pada perhitungan similaritas (T,S2) nilai 100% yang didapatkan palsu karena seharusnya masih terdapat komponen/atribut yang perlu menjadi pertimbangan dan tidak

masuk dalam perhitungan yaitu atribut **PR01** dan **PR02** yang ada pada kasus target sebenarnya juga memiliki nilai. Pada penelitian selanjutnya fungsi similaritas ini tidak hanya memperhatikan total dari nilai bobot tetapi juga memperhatikan jumlah atribut antara target (T) dan basis kasus (S), serta memperhatikan nilai pembagi yang digunakan, nilai pembagi yang digunakan seharusnya nilai maksimal diantara keduanya (target/basis kasus). Atau dengan menetapkan nilai “1” pada atribut yang “diabaikan”, sehingga kasus kemiripan palsu tidak terulang kembali.

## 5 Kesimpulan

Berdasarkan hasil analisis, perancangan serta pengujian yang dilakukan pada penelitian ini, berikut beberapa kesimpulan yang dapat diambil :

1. Kondisi awal dan keluhan pada sistem ini digunakan sebagai *problem case* sedangkan nama jenis masalah dan solusi digunakan sebagai *solution space*.
2. Nilai kemiripan antara kasus baru dengan kasus yang ada di basis kasus bernilai antara 0 atau 1. Nilai 0 berarti menunjukkan kemiripan kasus tidak ada sedangkan kemiripan dikatakan sama apabila kemiripan kasus baru dengan kasus yang ada di basis kasus bernilai 1 atau 100%.
3. Proses *retrieval* atau *similarity* dilakukan dengan cara membandingkan setiap atribut (kondisi awal dan keluhan) antara kasus baru dengan kasus yang ada di basis kasus, sedangkan perhitungan similaritas dilakukan dengan teknik *nearest neighbor*. Dari hasil perhitungan kemiripan *nearest neighbor* sistem akan menampilkan beberapa kasus yang mempunyai nilai similaritas diatas nilai *threshold*.

## 6 Pustaka

1. Althoff, K. D. (2001). *Case-Based Reasoning*, Handbook of Software Engineering & Knowledge Engineering (ed.S.K. Chang) Vol 1, World Scientific, Singapore.
2. Watson, I. (1997). *Applying Case-Based Reasoning, Technique for Enterprise Systems*, Morgan Kaufmann Publishers.
3. Waterman, D. A. (1986). *A Guide to Expert Systems*. Canada: Addison-Wesley Publishing Company.
4. Kusumadewi, Sri. (2003). *Artifical Intelligence (Teknik dan Aplikasinya)*. Graha Ilmu. Yogyakarta.
5. Giarratano, J, dan Riley. G., (1993). *Expert System*, University of Houston. Clear Lake and NASA. Johnson Space Center.
6. Sankar K. Pal dan Simon. (2004). *Foundations Of Soft Case-Based Reasoning*, Wiley-Interscience.
7. Kusrini and Hartati, S, (2011) *Penggunaan Penalaran berbasisi Kasus Untuk membangun Basis Pengetahuan dalam Sistem Diagnosa Penyakit*, [Http://dosen.amikom.ac.id/downloads/artikel/full\\_paper\\_sriti\\_kusrini.pdf](http://dosen.amikom.ac.id/downloads/artikel/full_paper_sriti_kusrini.pdf) , diakses pada tanggal 07 September 2015.
8. Kartono, Kartini. (1995). *Bimbingan Belajar di SMA dan Perguruan Tinggi*. Raja Grafindo Persada. Jakarta
9. Ardian Nur Romadhan. (2013). *Implementasi Case-Based Reasoning untuk Pendukung Dokter Jaga dalam Mendiagnosa Penyakit pada RSU PKU Muhammadiyah Delangg*. STMIK AMIKOM Yogyakarta.