Jurnal Ilmiah Informatika (JIF)
| ISSN (Print) 2337-8379 | ISSN (Online) 2615-1049 |
Jurnal online di akses di http://ejournal.upbatam.ac.id/index.php/jif

Sistem Deteksi Kerusakan pada Sistem Operasi Menggunakan Metode TF-IDF dan *Cosine Similarity*

Aa Zezen Zaenal Abidin [a], *Andi Sukmadinata* [b]

[a] *STMIK Subang, Jl.marsinu No 5 Subang, Subang 40123, Indonesia*
[b] *STMIK Subang, Jl.marsinu No 5 Subang, Subang 40123, Indonesia*

INFORMASI ARTIKEL

Sejarah Artikel
Diterima Redaksi: 25 Juni 2020
Revisi Akhir: 01 September 2020
Diterbitkan *Online*: 25 September 2020

KATA KUNCI

Text Mining
Sistem Deteksi
TF-IDF
Cosine similarity
Sistem Operasi

KORESPONDENSI

E-mail: zezen2008@yahoo.com
E-mail: andy93sukmadinata@gmail.com

A B S T R A C T

System damage to the operating system, errors in the operating system, with damage to software and hardware. The detection system is expected to be more flexible than an ordinary expert system, because in an ordinary expert system the consultation is guided while in the detection system using the text similarity method, the user can express the consultation using free expressions on the user consultation menu by using the user consultation text. The system uses the Term Frequency-Inverse Document Frequency method. Once the operating system malfunction query is filled in to the system, the query preprocessing is carried out and the text document is in the database, dedicating the weight of the relationship of a word to the document. After doing the word weighting process, then do the document crunching against the query using the Cosine Similarity method. A collection of text that has been classified in the database which is used as the basis of knowledge and the text consulted as a query, obtained the operating system damage detection system with two categories, namely software and hardware damage. The system is able to create consulted crashes by checking the similarity of the query text and knowledge base. The results of the evaluation using a matrix that shows an accuracy value of 70 percent, the next research in error detection using text similarity is expected to increase the reliability of the system with even greater assessments.

1. PENDAHULUAN

Implementasi *text mining* untuk memastikan kesalahan pada *operating system* dengan menggunakan *TF/IDF* dan *cosine similarity* merupakan suatu aplikasi yang dirancang untuk memudahkan pengguna dalam mengatasi masalah yang dialami saat menggunakan *operating system*. Dimana yang diprioritaskan dalam aplikasi ini adalah pencarian identifikasi masalah berdasarkan kemiripan teks. Pengetauan yang disajikan dalam representasi teks, *knowledge* di representasikan dalam bentuk teks.

Data mining atau sering juga disebut dengan *knowledge discovery in Database* (KDD) merupakan proses untuk menemukan *interesting knowledge* dari sejumlah besar data yang disimpan baik di dalam *database*, *datawarehouse* atau tempat informasi penyimpanan lainnya. *Data mining* merupakan salah satu tahap yang terdapat di *knowledge discovery*. Metode *TF-IDF (Term Frequency-Inverse Document Frequency)* merupakan suatu cara untuk memberikan bobot hubungan suatu kata (*term*) terhadapa dokumen. Metode ini akan mengabaikan setiap kata-kata yang tergolong tidak penting. Oleh sebab itu sebelum melakukan metode ini, proses *stemmer* harus dilakukan terlebih dahulu oleh *system*. *Cosine similarity* akan menenetukan nilai kemiripan dokumen keseluruhan dari query dan dokumen pengetahuan dalam *knowledge base*.

Beberapa bentuk implementasi dalam pengelolaan teks, diantaranya adalah kategorisasi teks[1], klasifikasi dokumen teks[2] seperti juga klasifikasi varian dokumen oleh [3], analisis *cross-language sentiment*[4], sistem pendeteksian kemiripan judul karya ilmiah[5], pencarian artikel dan klausa dalam dokumen UUD 1945[6], penentuan dosen penguji[7], metode pengelolaan teks digunakan untuk mendeteksi durasi waktu berita online[8]. Dalam penelitian ini khusus digunakan unutk mendeteksi kerusakan pada system operasi. Secara khusus kerusakan pada system operasi diasumsikan hanya dua saja, yaitu keursakan software dan kerusakan hardware.

2. TINJAUAN PUSTAKA

Pennggunaan kemiripan dokumen teks dilakukan oleh[9], diperoleh system yang dapat digunakan untuk mengecek kemiripan teks di system paten dengan nilai akurasi yang baik, dapat mengidentifikasi kemiripan paten anatar grup paten dari bebrapa perusahaan pemilik paten. Penggunaan cisine similarity dalam penelitian ini lebih baik disbanding dengan Euclidean dalam system temu kembali informasi klinik [10] demikain juga dengan [11] sama pada data klinik, khususnya data teks untuk kemiripan pasien. Menganalisis biogeografis dan ekologis asal dan pemilihan spesies arctic menggunakan teks mining dalam sebuah system *expert-knowledge*[12] penggunaan lainnya teks mining oleh[13]. Menggunakan metode pengeolahan teks untuk menganalisis data bio medis [14].

Text mining adalah salah satu bidang khusus dari data mining. Sesuai dengan buku *The Text Mining Handbook.* Salah satu elemen kunci dari *text mining* adalah kumpulan dokumen yang berbasis teks. Pada prakteknya, *text mining* ditujukan untuk menemukan pola dari sekumpulan dokumen yang jumlahnya sangat besar dan bisa mencapai jumlah ribuan bahkan sampai jutaan. Koleksi dokumen bisa statis, dimana dokumen tidak berubah, atau dinamis, dimana dokumen selalu di*update* sepanjang waktu.

Beberapa Teknik Pendekatan statistika adalah Sebagai berikut[15]:

a) Teknik *Word Frequency*
b) *Position in text*
c) *Cue Words and heading*
d) *Setence position*

Teknik Pendekatan dengan Naturan Language Analysis

a) Inverse Term Frequency and NLP Technique
b) Lexical chain,
c) Maximal Marginal Relevance

Salah satu metode yang sangat berguna dan efektif dalam system temu kembali informasi adalah metode vector space model (VSM), pembobotan term dari query maupun kaliamat dalam metode ini menggunakan *Term Frequency- Inverse Document Frequency*(TF-IDF)[16].

Dalam penelitian ini tentu yang terbaru adalah penggunaan dokumen yang tersimpan sebagai dokumen pengetahuan dalam basis data, biasanya disebut sebagai *Knowlegde Base Discovery* (KD), sedangkan query dijadikan sebagai konsultasi dari user kepada system. Tentu saja diharapkan dalam penelitin ini bahwa ekspresi konsultasi user kepada sistem bisa dilakukan secara lebih fleksibel, tidak kaku seperti pada system pakar yang menggunakan aturan secara terbimbing. Dalam metode ini pembobotan kata dalam sebuah dokumen dilakukan dengan mengalikan nilai Tf dan IDF. Pembobotan diperoleh berdasarkan jumlah kemunculan term dalam kalimat (TF) dan jumlah kemunculan term pada seluruh kalimat dalam dokum (IDF). Bobot suatu istilah semakin besar jika istilah tersebut sering muncul dalam suatu dokumen dan semakin kecil jika istilah tersebut muncul dalam banyak dokumen, Nilai IDF sebuah term dihitung menggunakan persamaan dibawah [15]:

Tahap selanjutnya adalah seleksi fitur, yaitu mencari nilai *tf-idf* dengan rumus

tf-idf = *tf* x log(N/*df*)……..1)

di mana,

tf : jumlah *term* tersebut
N : total dokumen
df : jumlah dokumen yang mengandung suatu *term*

Menghitung bobot (W) masing-masing dokumen dengan persamaan dibawah ini:

df=d1+d2+…+dn **…… (1)**

Keterangan pada rumus 1

- Df : Total kata pada term
- D : frekuensi kata pada kalimat

,id f = log (n/Df) **…… (2)**

Keterangan pada rumus 2

- N : total dokumen
- Df : banyak dokumen yang mengandung kata yang dicari

***W (t, d) = tf (t, d) * idf* ……..... (3)**

Keterangan pada rumus 3

- d : dokumen ke – d
- t : kata ke -t dari kata kunci
- tf : banyaknya kata yang dicari pada sebuah dokumen
- IDF : *Inversed Document Frequency*

Kemudian baru melakukan proses pengurutan (*sorting*) nilai kumulatif dari W untuk setiap kalimat. Tiga kalimat dengan nilai W tersebesar dijadikan sebagai hasil dari ringkasan atau output dari peringkasan teks otomatis. Pengujian system dilakukan menggunakan *Confusion matrix*[17], seperti pada Table 1[18], dimana *Confusion matrix* merupakan metode yang menyatakan jumlah data uji yang benar diklasifikasikan dan jumlah data uji yang salah diklasifikasikan.

Tabel 1 *Confusion Matrix* untuk klasifikasi biner

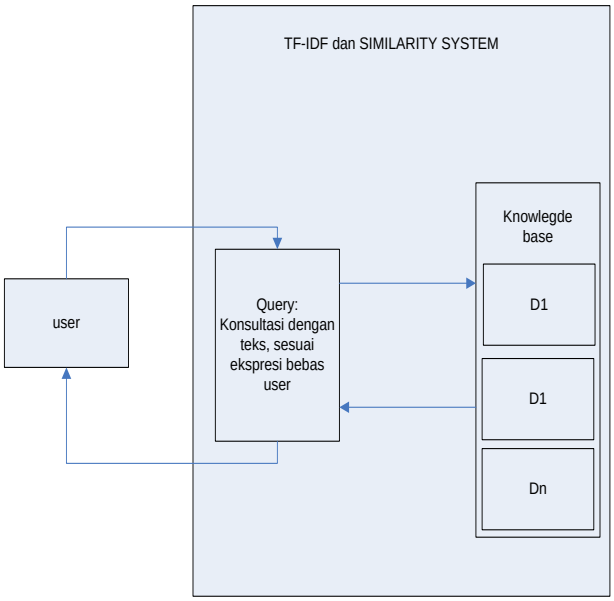
		Kelas Prediksi	
		1	0
Kelas Sebenarnya	1	TP	FN
	0	FP	TN

Keterangan Tabel 6 :

True Posstive (TP)	= jumlah dokumen dari kelas 1 yang benar dan diklasifikan sebagai kelas 1.
True Negative (TN)	= jumlah dokumen dari kelas 0 yang benar diklasifikasikan sebagai kelas 0.
False Positive (FP)	= jumlah dokumen dari kelas 0 yang salah diklasifikasikan sebagai kelas 1.
False Negative (FN)	= jumlah dokumen dari kelas 1 yang salah diklasifikasikan sebagai kelas 0.

3. METODOLOGI

Dalam penelitian ini *query* dijadikan sebagai representasi konsultasi dari pengguna, begitu pengguna melihat kerusakan pada sistem operasi, sedangkan dokumen dalam basis data dari D1 hingga Dn menjadi salah satu pengetahuan dalam *Knowledge Base* atau basis pengetahuan, seperti diperlihatkan dalam Gambar 1.



Gambar 1 Sistem *knowledge base* deteksi kerusakan sistem operasi

Terdapat dua dokumen masalah pada sistem operasi dan satu *query* yang mana akan dilakukan proses *pre-processin*, .seperti pada Tabel 2.

Tabel 2 Dokumen Uji

Dokumen	Isi Dokumen
Q	Menyalakan computer, monitor *blank*, hardisk mengeluarkan suara
D1	komputer mati total bunyi beep 1x 2x 3x 4x hardisk bunyi monitor blang insert system disk harddisk error failure operating system not found bad sector cd dvd room tidak jalan tidak terbaca jam dan tanggal selalu berubah keyboard mouse tampilan display rusak vga card sound lan connector power suplay ram
D2	komputer tidak mengeluarkan suara usb tidak terdeteksi windows teraktivasi not genuine Preparing Your Desktop internet Sistem32 *mising safe mode driver not found wifi* resolusi tidak sesuai not responding idm fake serial number google chrome fire fox *crashed devices media player mircosoft office system restore virtual memory* anti virus bluetooth manager prorgram files explorer desktop

Tahap selanjutnya adalah seleksi fitur, yaitu mencari nilai *tf-idf* dengan rumus

tf-idf = *tf* x log(N/*df*)……..1)

dari 81 *record* atau 81 *term*, hanya diambil 10 untuk dokumentasi ini, seperti pada Tabel 3.

Tabel 3 Dokumen setelah pemecahan kata

No	**Tf**				
	Term	**Q**	**D1**	**D2**	**Df**
1	*Computer*	1	1		2
2	Mati		1		1
3	Mengeluarkan	1		1	2
4	Monitor	1	1		2
5	Blang	1	1		2
6	Virus			1	1
7	Microsoft			1	1
8	Hardisk	1	1		2
9	Bluetooth			1	1
10	Suara	1		1	2

Selanjutnya adalah menghitung nilai *idf* dan *TF-IDF*., seperti pada Tabel 4.

Tabel 4 Nilai *TF-IDF*

Idf	**wdt=tf*idf**		
log(n/df)	**Q**	**D1**	**D2**
0,176091	0,176091	0,176091	0
0,477121	0	0,477121	0
0,176091	0,176091	0	0,176091
0,176091	0,176091	0,176091	0
0,176091	0,176091	0,176091	0
0,477121	0	0	0,477121
0,477121	0	0	0,477121
0,176091	0,176091	0,176091	0
0,477121	0	0	0,477121
0,176091	0,176091	0	0,176091

Rumus perhitungan mencari **idf** :

=LOG(N/df)
=LOG(3/2)
=0,845098…

Rumus perhitungan mencari **wdt** dar i **q** :

=q*idf
=1*0,845098
=0,845098…

Rumus perhitungan mencari **wdt** dari **d1**:

=d1*idf
=0*0,845098
=0

Rumus perhitungan mencari **wdt** dari **d2** :

=d2*idf
=0*0,845098
=0

Tabulasi hasil semua perhitungan diperlihatkan pada Tabel 5.

Tabel 5 Panjang Vektor

No	Wdq * Wdi		Panjang Vektor		
	D1		D2		
1	0	0	0,714191	0	0
2	0,13541	0,13541	0,135407	0,1354069	0,135407
3	0,29601	0	0,29601	0,29601	0
4	0,29601	0	0,29601	0,29601	0
5	0,29601	0	0,29601	0,29601	0
6	0	0	0	0	0,714191
7	0	0	0	0	0,714191
8	0	0	0	0,7141907	0
9	0	0	0	0,7141907	0
10	0	0	0	0,7141907	0

Setelah melakukan proses pembobotan kata, langkah selanjutnya melakukan perangkingan dokumen terhadap *query* dengan menggunakan metode *Cosine Similarity*.

$$\cos\mathrm{Sim}(\vec{d}_{\mathrm{j}},\vec{q}) = \frac{\vec{d}_{\mathrm{j}}.\vec{q}}{|\vec{d}_{\mathrm{j}}|.|\vec{q}|}$$

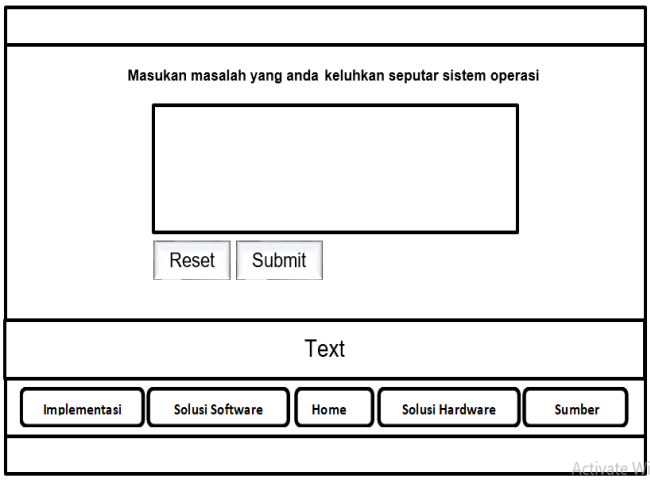
$$= \frac{\sum_{i=1}^{t}(w_{\mathrm{ij}} \cdot w_{\mathrm{iq}})}{\sqrt{\sum_{i=1}^{t} w_{ij}{}^2 \cdot \sum_{i=1}^{t} w_{ij}{}^2}}$$

Rekapan semua hasil perhitungan pada similarity pada penelitian ini disampaikan pada Tabel 6.

Tabel 6 Tahap perhitungan dengan *Cosine Similarity*

No	Wdq * Wdi		Panjang Vektor		
	D1	D2	Q	D1	D2
1	0,031008	0	0,031008	0,031008	0
2	0	0	0	0,227645	0
3	0	0	0,031008	0,031008	0,031008
4	0,031008	0	0,031008	0,031008	0
5	0,031008	0	0,031008	0,031008	0
6	0	0	0	0	0,227645
7	0	0	0	0	0,227645
8	0,031008	0	0,031008	0,031008	0
9	0	0	0	0	0,227645
10	0	0	0,031008	0,031008	0,031008
…	……	……	……	……	……
….	……	0,124033	0,031008	0,155041	0,351677
				0,393752	0,593024
					0,531179
					0,844951
					0,093201

Rancangan antar muka dari sistem diperlihatkan dalam Gambar 2, dimana terdapat tempat untuk menuliskan teks sebagai *query* yang dituliskan pengguna untuk melakukan konsultasi berdasarkan kondisi sistem operasi yang digunakan pada komputer dari pengguna.



Gambar 2 Rancangan antar muka sistem

4. HASIL DAN PEMBAHASAN

Implementasi antar muka system diperlihatkan dalam Gambar 3.

Gambar 3 Implementasi anatar muka system

Untuk menguji tingkat akurasi pada Implementasi *Text Mining* Untuk Memastikan Kesalahan Pada *Operating System* menggunakan metode *Cosine Similarity* adalah *Confusion matrix.*

Tabel 7 Data Hasil Klasifikasi Data Uji

No	Kelas Sebenarnya	Kelas *cosSim*
1	Kesalahan Software	Kesalahan Software
2	Kesalahan Hardware	Kesalahan Hardware
3	Kesalahan Software	Kesalahan Hardware
4	Kesalahan Hardware	Kesalahan Software
5	Kesalahan Software	Kesalahan Software
6	Kesalahan Hardware	Kesalahan Hardware
7	Kesalahan Software	Kesalahan Software
8	Kesalahan Software	Kesalahan Software
9	Kesalahan Hardware	Kesalahan Hardware
10	Kesalahan Software	Kesalahan Hardware

Setelah Data uji didapatkan, maka dilakukan tahap pengujian tingkat akurasi.

Tabel 8 Data hasil perhitungan mencari nilai TP, FP, FN dan TN

		Kelas Prediksi	
		1	0
Kelas Sebenarnya	1	4	2
	0	1	3

Keterangan Tabel 8 :
1 = Kesalahan *Software*
0 = Kesalahan *Hardware*

Dari tabel 8 maka diketahui nilai TP = 4, FN = 2, FP = 1, TN = 3. Langkah berikutnya yaitu menggunakan persamaan :

$$\text{Akurasi} = \frac{TP + TN}{TP + FN + FP + TN} \text{ X } 100\%$$

$$\text{Akurasi} = \frac{4 + 3}{4 + 2 + 1 + 3} \text{ X } 100\%$$

$$= 70\ \%$$

5. KESIMPULAN

Sistem deteksi kerusakan pada system operasi yang disederhanakan pada kategori kumpulan teks kerusakan software atau hardware dapat menggunakan metode teks mining TF-IDF dengan perolehen akurasi sebesar 70 persen. Kumpulan teks yang sudah diklasifikasikan dapat dijadikan basis pengatahuan, dengan metode kemiripan dapat konsultasi pengguna yang disampaikan oleh pengguna melalui *query* dapat diperoleh sistem deteksi. Sistem menghasilkan besaran akurasi yang belum terlalu memuaskan karena masih lebih kecil dari 80 persen, penelitian lebih lanjut bisa dilakukan dengan spesifikasi kerusakan pada system operasi lebih spesifik dan komplek dan pemilihan teks dokumen sebagai *knowledge base* yang lebih baik.

DAFTAR PUSTAKA

[1] R. Ju, P. Zhou, C. H. Li, and L. Liu, “An Efficient Method for Document Categorization Based on Word2vec and Latent Semantic Analysis,” 2015.

[2] B. Kuyumcu, B. BULUZ, and Y. KOMECOGLU, “Ridge Regresyon Analizi ile Türkçe Dokümanlarda Yazar Tanıma Author Identification in Turkish Documents with Ridge Regression Analysis,” pp. 0–3, 2019.

[3] A. Mishra and S. Vishwakarma, “Analysis of TF-IDF Model and its Variant for Document Retrieval,” pp. 772–776, 2015.

[4] H. Ma, Y. Zhang, and Z. Du, “Cross-language Sentiment Classification Based on Support Vector Machine *,” pp. 507–513, 2015.

[5] Nurdin and A. Munthoha, “SISTEM PENDETEKSIAN KEMIRIPAN JUDUL SKRIPSI MENGGUNAKAN,” *J. Nas. dan Teknol. Jar.*, vol. 2, pp. 90–97, 2017.

[6] O. R. Sulaeman, W. Gata, M. Wahyudi, R. Subandi, R. Setiyawan, and B. Pratama, “Information Retrieval System to Find Articles and Clauses in UUD 1945 Using Vector Space Model Method Information Retrieval System to Find Articles and Clauses in UUD 1945 Using Vector Space Model Method,” *J. Phys. Conf. Ser.*, 2020.

[7] R. R. A. Siregar, F. A. Sinaga, R. Arianto, P. Studi, S. Teknik, and K. Kunci, “APLIKASI PENENTUAN DOSEN PENGUJI SKRIPSI MENGGUNAKAN METODE TF-IDF DAN VECTOR SPACE MODEL,” *J. Comput. Sci. Inf. Syst.*, vol. 2, pp. 171–186, 2017.

[8] F. Wiranto, A. Maududie, and T. Dharmawan, “Time Frame Detection Based on Online News Documents Using Vector Space Model,” *2019 Int. Conf. Comput. Sci. Inf. Technol. Electr. Eng.*, vol. 1, no. 1, pp. 19–23, 2019.

[9] S. Arts, B. Cassiman, and J. C. Gomez, “Text matching to measure patent similarity,” *Strateg. Manag. J.*, vol. 39, no. 1, pp. 62–84, 2018.

[10] M. Laburu, A. Perez, A. Casillas, I. Goenaga, and M. Oronoz, “Can i find information about rare diseases in some other language?,” *Proc. - 2018 IEEE Int. Conf. Bioinforma. Biomed. BIBM 2018*, pp. 2102–2108,

2019.
[11] P. A. Hummel, F. Jakel, S. Lange, and R. Mertelsmann, “Case-based Reasoning Reaseach and Development,” vol. 11156, pp. 264–280, 2018.
[12] M. H. Hoffmann, “TEXT MINING OF EXPERT KNOWLEDGE FOR THE CONSTRUCTION OF A GLOBAL HABITAT SPACE OF MICRANTHES AND SAXIFRAGA REVEALS MULTIPLE AVENUES OF ARCTIC BIOME ASSEMBLY,” vol. 180, no. 3, 2019.
[13] E. Da Costa, H. Tjandrasa, and S. Djanali, “Text mining for pest and disease identification on rice farming with interactive text messaging,” *Int. J. Electr. Comput. Eng.*, vol. 8, no. 3, pp. 1671–1683, 2018.
[14] G. Soganc ıoglu, H. Oztu, and A. Ozgu, “BIOSSES : a semantic sentence similarity estimation system for the biomedical domain,” no. March, 2018.
[15] M. Mustaqhfiri, Z. Abidin, and R. Kusumawati, “Peringkasan Teks Otomatis Berita Berbahasa Indonesia Menggunakan Metode Maximum Marginal Relevance,” *Matics*, no. March 2012, 2012.
[16] S. Jabri, A. DAHBI, and T. GADI, “Ranking of Text Documents using TF-IDF Weighting and Association Rules mining,” 2018.
[17] M. Deshpande and V. Rao, “Depression detection using emotion artificial intelligence,” *Proc. Int. Conf. Intell. Sustain. Syst. ICISS 2017*, no. Iciss, pp. 858–862, 2018.
[18] C. J. Rameshbhai and J. Paulose, “Opinion mining on newspaper headlines using SVM and NLP,” *Int. J. Electr. Comput. Eng.*, vol. 9, no. 3, pp. 2152–2163, 2019.

BIODATA PENULIS

Aa Zezen Zaenal Abidin, S.Pd, S.T.,M.Cs
Dosen Departemen Informatika STMIK Subang, lulusan Magister Ilmu Komputer UGM, memiliki fungsional lektor..

Andi Sukmadinata
Lulusan STMIK Subang tahun, praktisi IT sejak lulus kuliah dari STMIK Subang..