PENGGUNAAN COSINE SIMILARITY UNTUK MENCARI KESAMAAN KANDUNGAN OBAT

Kristophorus Hadiono [1], Heribertus Yulianton [2], Dwi Agus Diartono [3]
[1.3]Program Studi Sistem Informasi, Fakultas Teknologi Informasi, Universitas Stikubank
[2]Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Stikubank
e-mail: [1]kristophorus.hadiono@edu.unisbank.ac.id, [2]heri@edu.unisbank.ac.id,
[3]dwieagus@edu.unisbank.ac.id

ABSTRAK

Mengingat banyaknya informasi berbasis teks yang dapat disimpan, memunculkan potensi kesulitan untuk mendapatkan informasi yang diperlukan. Dampaknya, membutuhkan waktu lama jika mencari dokumen satu demi satu. Untuk itu, diperlukan cara mengakses informasi secara cepat dan tepat. Pencarian kata adalah salah satu bagian dari Information Retrieval, salah satu modelnya adalah Vector space model. VSM memiliki efektifitas dalam pencarian kata karena hasil pencariannya berdasarkan kemiripan vectorquery dan vector dokumen. Penelitian ini mengimplementasikan Algoritma VSM dengan tahapan : preproccessing text menggunakan 4 tahapan, pembobotan term menggunakan metode TF-IDF, dan perangkingan menggunakan metode Cosine Similarity.

***Kata Kunci**: vector space model, tf-idf, cosine similarity*

1. PENDAHULUAN

Pencarian kata merupakan salah satu bagian dari *Information Retrieval*. *Information Retrieval* merupakan bagian dari computer science yang berhubungan dengan pengambilan informasi dari dokumen-dokumen yang didasarkan pada isi dan konteks dari dokumen-dokumen itu sendiri. *Information Retrieval* yaitu ilmu pencarian informasi pada dokumen, pencarian untuk dokumen itu sendiri, mencari di dalam database untuk teks, suara, gambar, atau data lainnya.

Ada tiga model yang digunakan dalam *information retrieval*, yang pertama *Probabilistic model*, contoh model ini ialah penerapan teorema *Bayes* dalam model *probabilistic*, yang kedua *Set-theoretic models*, contoh model ini ialah *Standard boolean* dan yang terakhir *Algebratic model*, contoh model ini adalah merepresentasikan dokumen dan *query* sebagai *vector similarity* antara *vector* dokumen dan *vector query*. Contoh model ini ialah *Vector space model.* Dari tiga model *information retrieval*, *Algebratic model* dengan contoh model *Vector space model* adalah model yang paling sederhana dalam pencarian kata, telah terbukti memiliki efektifitas dalam pencarian kata dengan menampilkan hasil pencariannya berdasar kemiripan *vector query* dan *vector* dokumen.

Vector Space Model merupakan model IR yang merepresentasikan dokumen dan *query* dalam bentuk vektor dimensional. Konsep dasar dari VSM adalah menghitung jarak antar dokumen kemudian mengurutkan berdasarkan tingkat kedekatannya. Semakin kecil jarak antar dokumen, maka semakin mirip keduanya (Bari & Saputra, 2011, hal. 3-4).

Tujuan khusus yang diharapkan dari penelitian ini adalah dapat menggunakan metode Vector Space Mode untuk mengukur kesamaan kandungan obat.

2. TINJAUAN PUSTAKA

2.1 Sistem Temu Kembali Informasi

Sistem temu kembali informasi merupakan kegiatan yang bertujuan untuk menyediakan dan memasok informasi bagi pemakai sebagai jawaban atas permintaan atau berdasarkan kebutuhan pemakai. Prinsip kerja sistem temu kembali informasi jika ada sebuah kumpulan dokumen dan seorang pengguna yang memformulasikan sebuah pertanyaan (request atau query). Jawaban dari pertanyaan tersebut adalah sekumpulan dokumen yang relevan dan membuang dokumen yang tidak relevan.

Untuk dapat melakukan pencarian berdasar substansi yang paling mirip, terdapat teknologi yang disebut Information Text Retrieval. Information text retrieval adalah salah satu metode yang digunakan untuk menyimpan data dengan cara memprosesnya (menghilangkan stop word) dan menyimpan tiap kata

beserta informasi dari kata tersebut (letak kata, jumlah bobot, dll). Information retrieval berfokus pada proses yang terlibat di dalam representasi, media penyimpanan, mencari dan menemukan informasi yang relevan dari informasi yang diinginkan oleh user (Ingwersen, 1992)

2.2 Vector Space Model

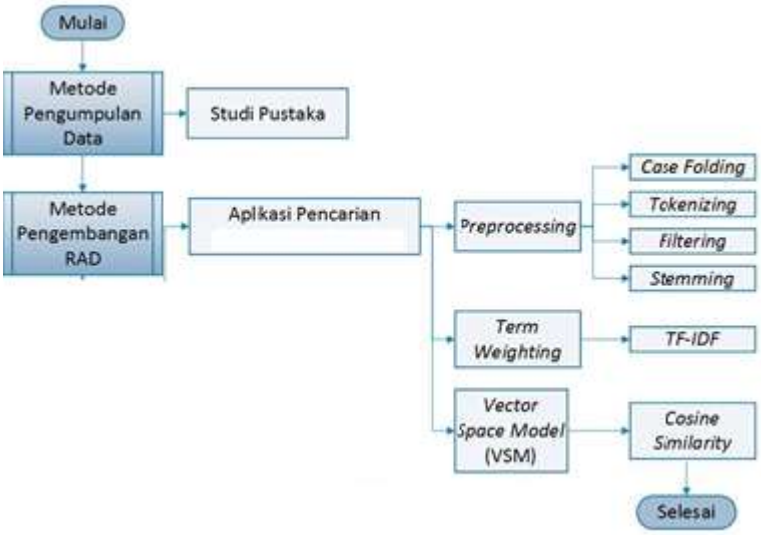
Fauzia Tri Mumpuni (2017) dalam penelitiannya mengemukakan bahwa implementasi VSM untuk pencarian kata pada terjemahan Al-Qur'an melalui tahapan pertama yaitu proses text preproccessing yang menggunakan 4 tahapan yaitu case folding, tokenizing, filtering, stemming (algoritma porter stemmer) dan term weighting (metode TF-IDF) yang berfungsi untuk memaksimalkan hasil pencarian. Dan proses similaritas (cosine similarity) yang berfungsi untuk mendapatkan kecocokan ayat dengan query serta jaraknya untuk pengurutan.

VSM memiliki efektifitas dalam pencarian kata karena hasil pencariannya berdasarkan kemiripan vector query dan vector dokumen. Penelitian ini mengimplementasikan Algoritma VSM dengan tahapan : preproccessing text menggunakan 4 tahapan, pembobotan term menggunakan metode TF-IDF, dan perangkingan menggunakan metode Cosine Similarity.

Menurut Baeza (dalam Amin, 2012:80) *Vector Space Model (VSM)* adalah metode untuk melihat tingkat kedekatan atau kesamaan *(similarity) term* dengan cara pembobotan *term*. Dokumen dipandang sebagai sebuah vektor yang memiliki *magnitude* (jarak) dan *direction* (arah). Pada *Vector Space Model*, sebuah istilah direpresentasikan dengan sebuah dimensi dari ruang vektor. Relevansi sebuah dokumen ke sebuah *query* didasarkan pada similaritas diantara vektor dokumen dan vektor *query.*

3. METODE PENELITIAN

Penelitian yang dilakukan adalah mengambil data obat sebanyak 30, dan kemudian dipisahkan kata per kata untuk mendapatkan komposisi obatnya dan langkah berikutnya adalah diproses untuk mendapatkan hasil cosine similirity. Adapun tahapan yang dilakukan dapat dilihat dari gambar 1. dibawah ini.



Gambar 1. Analisa Sistem

4. HASIL DAN PEMBAHASAN

Indikator yang digunakan adalah data-data obat beserta komposisinya sebanyak 30. Dan tujuan yang ingin dicapai adalah untuk mengetahui kesamaan kandungan obat pada masing-masing data obat dengan menggunakan VSM. Berikut data obat beserta kandungannya dapat dilihat pada tabel 1.

Tabel 1. Contoh data obat beserta kandungannya

DATA OBAT
1. ANAKONIDIN 30 ML
· Dextromethrophan

· Guaifenesin
· Pseudoephedrine HCl
· Chlorpheniramine Maleate
2. ASIFIT
· Ekstrak daunkatuk
· vitamin B1
· vitamin B2
· vitamin B6
· vitamin B12
3. Becom-C
· Vitamin B1 50 mg
· Vitamin B2 25 mg
· Vitamin B6 10 mg
· Vitamin B12 5 mcg
· Nikotinamida 100 mg
· Vitamin C 500 mg
· Panthothenic acid 18,4 mg.

4.1. Case Folding

Case Folding adalah tahapan pertama, yaitu menghilangkan karakter selain huruf dan mengubah semua *term* ke dalam huruf kecil.

Tabel. 2. Contoh Case Folding

CASE FOLDING
paracetamol
ibnuprofen
dextromethrophan
guaifenesin
pseudoephedrine HCl
chlorpheniramine Maleate
ekstrak daun katuk
vitamin B1
vitamin B2
vitamin B12
vitamin B6
vitamin c
nikotinamida
panthothenic acid
klordiazepoksida
klidinium Bromida
vitamin A
vitamin D3

4.2. Tokenizing

Tokenizing adalah proses memecah setiap term ke dalam array yang dipisahkan oleh spasi.

Tabel. 3. Contoh Tokenizing

TOKEN
paracetamol
ibnuprofen
dextromethrophan
guaifenesin
pseudoephedrine HCl
chlorpheniramine Maleate
ekstrak daun katuk
vitamin B1
vitamin B2
vitamin B12
vitamin B6
vitamin c
nikotinamida
panthothenic acid
klordiazepoksida
klidinium Bromida
vitamin A
vitamin D3
d-panthenol
l-lysine HCl
niacinamide
glukosa anhydrat
sodium chloride

4.3. Filtering dan Stemming

Filtering adalah proses menghapus kata yang terdapat dalam *database stoplist*. *Stemming* adalah tahapan yang membuat sebuah term menjadi bentuk kata dasarnya, dalam penelitian ini penulis menggunakan Algoritma Porter Stemmer karena waktu yang dibutuhkan lebih singkat dibandingkan dengan Algoritma Nazief & Adriani, aturan Algoritma Porter Stemmer dijelaskan dengan detail dalam jurnal *A Study of Stemming Effects on Information Retrieval in* Bahasa Indonesia oleh Fadillah Z Tala.

Tabel. 4. Contoh Filtering dan Stemming

NO	Token
1	Token
2	
3	Paracetamol
4	Ibnuprofen
5	Dextromethrophan

6	Guaifenesin
7	pseudoephedrine HCl
8	chlorpheniramine Maleate
9	ekstrak daun katuk
10	vitamin B1
11	vitamin B2
12	vitamin B12
13	vitamin B6
14	vitamin c
15	Nikotinamida
16	panthothenic acid
17	Klordiazepoksida
18	klidinium Bromida
19	vitamin A
20	vitamin D3

4.4 Term Weighting &Vector Space Model (VSM)

Dalam penelitian ini menggunakan query untuk paracetamol dan ibuprofen.

Menghitung *Term Frequency* (TF)

Dari hasil s*temming*, dilakukan penghitungan TF (*term frequency*) dan DF (*document frequency*). Yaitu TF merupakan jumlah suatu term pada setiap dokumen, dan DF adalah jumlah dokumen yang memiliki suatu term.

Tabel 5.Contoh Perhitungan TF & DF

Token	Q	TF						df
		D1	D2	D3	D4	D5	D6	
paracetamol	1							8
Ibnuprofen	1							1
dextromethrophan		1						1
guaifenesin		1						1
pseudoephedrine HCl		1						1
chlorpheniramine Maleate		1						4
ekstrak daun katuk			1					4
vitamin B1			1	1	1		1	11
vitamin B2			1	1			1	9
vitamin B12			1	1	1		1	10
vitamin B6			1		1		1	10
vitamin c				1			1	7
nikotinamida				1				1
panthothenic acid				1				1
klordiazepoksida						1		1
klidinium Bromida						1		1
vitamin A							1	3

vitamin D3							1	4
d-panthenol							1	1
l-lysine HCl							1	2
niacinamide							1	3

Menghitung *Document Frequency* dan *Inverse Document Frequency* (IDF)

Setelah didapatkan hasil perhitungan DF dilanjutkan dengan perhitungan IDF. Dalam perhitungan ini dibutuhkan nilai N yaitu total jumlah obat yang digunakan yaitu 30 data.

Tabel 6. Contoh Perhitungan IDF

Token	Q	TF						df	D/df	Idf
		D1	D2	D3	D4	D5	D6			
paracetamol	1							8	3,75	0,574031
ibnuprofen	1							1	30	1,477121
dextromethrophan		1						1	30	1,477121
guaifenesin		1						1	30	1,477121
pseudoephedrine HCl		1						1	30	1,477121
chlorpheniramine Maleate		1						4	7,5	0,875061
ekstrak daun katuk			1					4	7,5	0,875061
vitamin B1			1	1	1		1	11	2,72727273	0,435729
vitamin B2			1	1			1	9	3,33333333	0,522879
vitamin B12			1	1	1		1	10	3	0,477121
vitamin B6			1		1		1	10	3	0,477121
vitamin c				1			1	7	4,28571429	0,632023
nikotinamida				1				1	30	1,477121
panthothenic acid				1				1	30	1,477121
klordiazepoksida						1		1	30	1,477121
klidinium Bromida						1		1	30	1,477121
vitamin A							1	3	10	1
vitamin D3							1	4	7,5	0,875061
d-panthenol							1	1	30	1,477121
l-lysine HCl							1	2	15	1,176091
niacinamide							1	3	10	1
glukosa anhydrat								1	30	1,477121

Menghitung TF-IDF

Setelah didapatkan hasil perhitungan IDF dilanjutkan dengan perhitungan TF-IDF. Dalam perhitungan ini adalah mengkalikan hasil TF dengan hasil IDF, didapatkan 2 nilai yaitu bobot pada setiap kata dan bobot pada setiap kata.

Tabel 7. Contoh Perhitungan TF-IDF

Token	Q	TF		df	D/df	Idf	W		
		D1	D2				Q	D1	D2
paracetamol	1			8	3,75	0,574031	0,574031	0	0
ibnuprofen	1			1	30	1,477121	1,477121	0	0

dextromethrophan		1		1	30	1,477121	0	1,477121	0
guaifenesin		1		1	30	1,477121	0	1,477121	0
pseudoephedrine HCl		1		1	30	1,477121	0	1,477121	0
chlorpheniramine Maleate		1		4	7,5	0,875061	0	0,875061	0
ekstrak daun katuk			1	4	7,5	0,875061	0	0	0,875061
vitamin B1			1	11	2,72727273	0,435729	0	0	0,435729
vitamin B2			1	9	3,33333333	0,522879	0	0	0,522879
vitamin B12			1	10	3	0,477121	0	0	0,477121
vitamin B6			1	10	3	0,477121	0	0	0,477121
vitamin c				7	4,28571429	0,632023	0	0	0
nikotinamida				1	30	1,477121	0	0	0
panthothenic acid				1	30	1,477121	0	0	0
klordiazepoksida				1	30	1,477121	0	0	0
klidinium Bromida				1	30	1,477121	0	0	0
vitamin A				3	10	1	0	0	0
vitamin D3				4	7,5	0,875061	0	0	0
d-panthenol				1	30	1,477121	0	0	0
l-lysine HCl				2	15	1,176091	0	0	0
niacinamide				3	10	1	0	0	0
glukosa anhydrat				1	30	1,477121	0	0	0

Menghitung *Cosine Similarity*

Dalam proses ini, adalah mencari nilai cosine similarity. Adapun contoh perhitungan dapat dilihat pada tabel 8.

Tabel 8. Contoh perhitungan cosine similarity

cosine similirity		
0,108578	0,108578	0,108578
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
HASIL COSINE SIMILARITY		
D13	D16	D21
0,108578	0,108578	0,108578
0,034866	0,050224	0,043234
3,486577	5,02243	4,32341
3	1	2

5. KESIMPULAN

Dari penelitian yang telah dilakukan dapat disimpulkan beberapa hal sbb :

1. Dari hasil penelitian mendapatkan kata dasar untuk obat sebanyak 82 dan sebagai query nya adalah ibuprofen dan paracetamol.
2. Hasil penelitian penulis menyimpulkan bahwa metode VSM untuk pencarian kata untuk mengetahui komposisi atau kandungan obat melalui tahapan pertama yaitu proses *text preproccessing* yang menggunakan 4 tahapan yaitu *case folding*, *tokenizing*, *filtering*, *stemming* (algoritma porter stemmer) dan *term weighting* (metode TF-IDF) yang berfungsi untuk memaksimalkan hasil pencarian. Dan proses terakhir adalah similaritas (*cosine similarity*) yang berfungsi untuk mendapatkan kecocokan judul skripsi dengan query serta jaraknya untuk pengurutan.
3. Hasil pengukuran komposisi obat dengan menggunakan query paracetamol dan ibuprofen yang mempunyai nilai kemiripan untuk masing-masing obat decolgen dengan nilai kemiripan 0,05, obat feminax dengan nilai kemiripan 0,04 dan obat calostusin kablet dengan nilai kemiripan 0,03.

DAFTAR PUSTAKA

[1] Bari, A., & Saputra, R. H. (2011). *Penerapan Pencarian Kata dengan Vector Space Model pada Aplikasi Terjemahan Juz Amma berbasis Java ME*. Palembang: Prodi Teknik Informatika, STMIK GI MDP.

[2] Brata, D., & Hetami, A. (2015). *Perancangan Information Retrieval (IR) Untuk Pencarian Ide Pokok Teks Artikel Berbahasa Inggris Dengan Pembobotan Vector Space Model*. Stmik Asia Malang.

[3] Fitri, M. (2013). *Perancangan Sistem Temu Balik Informasi dengan Metode Pembobotan Kombinasi TF-IDF untuk Pencarian Dokumen Berbahasa Indonesia*. Pontianak: Prodi Teknik Informatika, Jurusan Teknik Elektro Fakultas Teknik, Universitas Tanjungpura.

[4] Fauziah Tri M (2017). Implementasi Pendekatan VSM Pada Pencarian Terjemahan AlQuran Juz 1-13 Berbasis Web, Prodi Teknik Informatika, UIN Jakarta

[5] Ingwersen, P, 1992, *Information retrieval Interaction*, *London*, Taylor Graham Publishing. http://www.db.dk/pi/iri [29 Agustus 2005]

[6] Harjanto, D. S., Endah, S. N., & Bahtiar, N. (2012). *Sistem Temu Kembali Informasi pada Dokumen Teks Menggunakan Metode Term Frequency Inverse Document Frequency (TF-IDF)*. Semarang: Universitas Diponegoro, Fakultas Sains dan Matematika, Jurusan Matematika & Jurusan Ilmu Komputer/Informatika.