TRANSFORMTIKA, Vol.17, No.2, Januari 2020, pp. 124 - 132
ISSN: 1693-3656,
journals.usm.ac.id/index.php/transformatika

Implementasi Algoritma *Cosine Similarity* pada sistem arsip dokumen di Universitas Islam Sultan Agung

Dedy Kurniadi[1], Sam Farisa Chaerul Haviana[2], Andika Novianto[3]
[1]Universitas Islam Sultan Agung/Jurusan Teknik Informatika
Jl.Raya Kaligawe KM.4, telp: 024-6583584, e-mail: ddy.kurniadi@unissula.ac.id
[2]Universitas Islam Sultan Agung/Jurusan Teknik Informatika
Jl.Raya Kaligawe KM.4, telp: 024-6583584, e-mail: sam@unissula.ac.id
[3]Universitas Islam Sultan Agung/Jurusan Teknik Informatika
Jl.Raya Kaligawe KM.4, telp: 024-6583584, e-mail: andika@unissula.ac.id

ARTICLE INFO

Article history:

Received 20 September 2019
Received in revised form 19 December 2019
Accepted 02 January 2020
Available online 31 January 2020

ABSTRACT

Archiving in University that have not been well organized will cause a problems, the documents need for structuring and archives properly in the systems for the good standard a universities. The most importance of ease in finding the required archives is an important reason why it is necessary to develop an archive search system that can facilitate and improve the process of searching the archived document. Apllying cosine similarity algorithm in Information Systems is a solution for University to organizing archived documents, results from this reserach is the systems can show the relavant document from database list with precision 88.8% and recall 76.1% from all the data in database.

Keywords: Cosine Similarity, Information Systems, precision, recall

1. Introduction

Arsip mulanya berasal dari bahasa yunani, bahasa yang digunakan sebagai penyebutan arsip adalah "*Archivum*" dari kata Archivum kemudian berkembang lagi menjadi lebih spesifik didalam bahasa belanda yang disebutkan dengan kata "*Archief*" yang berarti wadah atau tempat yang digunakan untuk menyimpan catatan-catatan atau gambar-gambar yang digunakan sebagai bahan untuk mengingat kembali [1].

Berkembangnya zaman metode penggunaan arsip mulai diterapkan pada media elektronik, penggunaan sistem elektronik bisa mempermudah dalam pengarsipan sebuah dokumen namun dengan banyaknya data yang ada perlu diperhatikan dokumen-dokumen yang relevan dengan yang dicari sehingga tidak menimbulkan data sampah dan menemukan dokumen yang berkualitas [2].

Pengelolaan yang dilakukan di Biro Administrasi Umum Universitas Islam Sultan Agung Semarang, masih diolah dengan secara konvensional, hal tersebut sangat berpotensi mengakibatkan berbagai macam masalah, dan masalah yang sering terjadi adalah seperti dokumen hilang, rusak dan terbengkalai. Dengan pengelolaan secara konvensional seperti ini juga akan menimbulkan pemborosan pada resource lembaga [12].

Sistem informasi kearsipan ini diterapkan di Biro Administrasi Umum UNISSULA yang mana pada fungsi pencariannya sistem informasi ini menggunakan Algoritma *Cosine Similarity*,

Received September 20, 2019; Revised December 19, 2019; Accepted January 02, 2020

salah satu algoritma yang bisa menangani dokumen sebagai solusi dari model pengelolaan arsip dengan banyak data secara terkomputerisasi agar pencarian datanya bisa relevan ketika diproses, sistem ini mencari arsip digital dengan menghitung bobot teks judul dokumen dan isi dokumen kemudian akan dibobotkan dan dicocoknya dengan data yang ada di database.

2. Research Method

Dalam memodelkan dokumen-dokumen berbentuk teks terdapat beberapa cara. Misalnya, data teks bisa dijadikan atau direpresentasikan dalam wadah data kata atau sebagai kantong-kantong kata, di mana kata-kata yang dicari dan yang akan diasumsikan muncul secara independen dalam bentuk urutan yang tidak material. Model seperti ini yaitu model dengan menggunakan kantong kata banyak dimanfaatkan dan digunakan dalam pengambilan informasi dan teks mining [3] [4].

Dalam kearsipan, database pasti akan penuh dengan *record* data dokumen arsip dengan jumlah yang besar. Jumlah data tersebut akan menimbulkan data-data di dalam database tersebut memiliki kemiripan antar dokumen yang satu dengan yang lainnya. Hal tersebut bisa diselesaikan dengan menerapkan data mining, konsep data mining dalam pencarian dokumen menggunakan *cosine similarity* terdiri dari proses Stemming menghilangkan imbuhan, kemudian proses stopword mengidentifikasi kata dasar, dan kemudian dilakukan tokenisasi melakukan penghitungan kata dasar yang muncul di suatu dokumen, kemudian meghitung bobot kata dari tiap dokumen, dan kemudian langkah yang terakhir adalah menghitung vektor yang dihasilkan dari proses penghitungan dari tiap kata dan antar kata yang dicari data dokumen, apabila jumlah vektornya mendekati nilai cos 1 maka tingkat kemiripannya semakin tinggi [5] .

Metode algoritma *cosine similarity* sendiri adalah salah satu metode yang bisa dimanfaatkan sebagai metode pencarian data di dalam data mining dan sering digunakan untuk mendeteksi dokumen-dokumen yang mirip, *cosine similarity* akan menghitung tingkat kesamaan antar dua buah atau lebih dari objek yang dinyatakan dalam vektor jumlahnya ada dua vektor dengan menggunakan kata kunci (*cosine*) [6] [7]. Jadi *cosine similarity* dapat digunakan untuk menemukan kemiripan dokumen dalam data set dengan jumlah yang besar dan dapat lebih cepat dan sesuai menemukan dokumen yang dicari.

Fungsi *Cosine Similarity* dapat dinyatakan dengan rumus:

$$Similarity\ (X,Y) = \frac{|X \cap Y|}{|X|^{\frac{1}{2}}.|Y|^{\frac{1}{2}}}$$

Dimana: z

$|X \cap Y|$ adalah jumlah kata yang ada pada dokumen X dan dokumen Y

$|X|$ adalah jumlah kata yang ada pada dokumen X

$|Y|$ adalah jumlah kata yang ada pada dokumen Y

Tahapan dalam melakukan implementasi *Cosine Similarity*, terdiri dari beberapa tahapan-tahapan yaitu, Stemming merupakan tahapan untuk menemukan kata dasar dari sebuah kata. Tahapan atau proses stemming merupakan proses menghilangkan imbuhan awalan, sisipan dan akhiran untuk mendapatkan kata dasar, setelah tahapan atau proses steming dilakukan, tahapan selanjutnya adalah menentukan indeksasi/ tokenisasi, proses tersebut melakukan penghitungan jumlah kata yang ada di dalam dokumen dan menghitung TF (*Term Frequency*), DF (*Document Frequency*), dan IDF (*Invers Document Frequency*) pada dokumen tersebut [8] [9] menggunakan rumus berikut :

$$\text{idf} = \log(\text{n}/\text{df})$$

dimana:

idf = *Invers Document Frequency*
n = jumlah term/kata tiap dokumen (*Term Frequency*)
df = *Document Frequency*

Setelah melakukan indeksasi/tokenisasi, dan menemukan TF (*Term Frequency*), DF (*Document Frequency*), dan IDF (*Invers Document Frequency)* pada setiap term, maka selanjutnya dilakukan perhitungan bobot WDT (*Weight Document Term*) dari setiap term [10] yang ada dengan menggunakan rumus:

(wdt) BOBOT [i] = tf[i] x idf[i]

dimana:

tf[i] : Term pada index i
idf[i] : IDF pada index i

Setelah selesai melakukan perhitungan WDT (*Weight Document Term*), maka selanjutnya melakukan perhitungan perkalian vektor antara Vektor Q dan Vektor D pada tiap term [11]. Vektor Q adalah hasil perhitungan wdt pada kata kunci dan Vektor D merupakan hasil perhitungan wdt pada tiap dokumen, Perkalian Vektor Q dan Vektor D dapat dinyatakan dengan rumus:

$$Q.D[i] = \sum_{j=1}^{n} wdt(Q_j) \times wdt(D[i]_j)$$

dimana :

wdt(Q) : Bobot Q pada kata ke j
D[i]j : Bobot D ke I pada Kata ke j

Apabila hasil perkalian Vektor Q dan Vektor D telah ditemukan, maka langkah selanjutnya yang dilakukan adalah menghitung panjang Vektor tiap term per dokumen yaitu dengan rumus:

$$|Q| = \sqrt{\sum_{j=1}^{n} wdt(Q)^2}$$

Setelah ditemukan hasil panjang vektor maka langkah terakhir yaitu menemukan hasil dari perhitungan *Cosine Similarity* yaitu menggunakan rumus :

$$similarity = \cos = \frac{Q.D_i}{|Q||D_i|}$$

dimana:

$Q.D_i$: Hasil perkalian vektor
$|Q||D_i|$: Hasil panjang vektor

3. Results and Analysis

Penggunaan Algoritma *Cosine Similarity* yang menentukan hasil terbaik dalam suatu pencarian, melalui beberapa proses, tahapan pertama melakukan stemming dari data yang ada pada database proses *stemming* menemukan kata dasar dengan menggunakan *stopword*. *Stopword* merupakan kata umum yang biasanya muncul dalam jumlah besar dan dianggap tidak memiki makna, seperti "yang", "di", "ke: dan lain lain. Proses *stemming* diawali dengan melakukan proses

menghilangkan imbuhan, awalan dan akhiran dari proses tersebut kemudian ditemukan kata dasar dari suatu kata di dalam dokumen yang di masukkan dalam kata kunci pencarian ataupun kata kunci dari judul dokumen. Berikut merupakan sampel data dalam melakukan proses *stemming* ditunjukkan pada tabel 3.1.

Tabel 3.1 Data set Arsip pada database.

	Judul	Isi
Q	Pemberhentian	
D1	PEMBERHENTIAN DAN PENGANGKATAN ANGGOTA SENAT UNISSULA 2014-2018	SK dengan Dasar Pengunduran Diri Prof. Choliq Dahlan dan Dr. Ibnu Khajar serta penambahan Prof. Widodo
D2	PEMBERHENTIAN DAN PENGANGKATAN KETUA PROGRAM MAGISTER TEKNIK ELEKTRO	Mengangkat Saudara Much Imam Ibnu Subroto sebagai Kaprodi MTE
D3	PEMBERHENTIAN KEPALA UPT SISTEM PENGENDALIAN MUTU INTERNAL (SPMI)	SK rektor ini memberhentikan dengan hormat Saudara Ahmad Salim, SE sebagai Kepala UPT SPMI terhitung mulai tanggal 31 Mei 2014
D4	Pemberhentian dengan hormat karena mencapai batas usia purna tugas an sdr Achmad Farouq	Surat YBWSA ini menerangkan Pemberhentian dengan hormat karena mencapai batas usia purna tugas an sdr Achmad Farouq terhitung mulai tanggal 1 Mei 2014
D5	Pemberhentian dengan hormat karena mencapai batas usia purna tugas sebagai Tenaga Pengajar Tetap Fakultas Kedokteran UNISSULA	SK YBWSA ini menerangkan Pemberhentian dengan hormat karena mencapai batas usia purna tugas sebagai Tenaga Pengajar Tetap Fakultas Kedokteran UNISSULA an dr. Hj. Utari terhitung mulai tanggal 1 Mei 2014
D6	Pemberhentian dengan hormat karena mencapai batas usia purna tugas sebagai Tenaga Pengajar Tetap Fakultas Kedokteran UNISSULA	SK YBWSA ini menerangkan Pemberhentian dengan hormat karena mencapai batas usia purna tugas sebagai Tenaga Pengajar Tetap Fakultas Kedokteran UNISSULA an dr. H. Muktasim Billah, Sp.S terhitung mulai tanggal 1 Februari 2014

Setelah melakukan stemming selanjutnya dilakukan Praprosessing terhadap judul dan isi
1. Judul dan isi digabungkan
2. Hapus Tanda Baca
3. Hapus Kata Bantu, Kata Ganti Orang, dll (*Stopword*)
4. Hapus imbuhan pada Kalimat (*Stemming*)

Setelah dilakukan proses *stemming* dan praprosessing maka ditemukan hasil Praprosessing dapat dilihat pada Tabel 4.2.

Tabel 2.2 hasil dari praprosessing.

Term	Data hasil *praprosessing*
Q	Henti
D1	henti hormat capai batas usia purna tugas sdr achmad farouq surat ybwsa erang henti hormat capai batas usia purna tugas sdr achmad farouq hitung tanggal mei
D2	henti kangkat anggota senat unissula sk dasar undur prof choliq dahl dr ibnu khajar tambah prof widodo
D3	henti kangkat ketua program magister teknik elektro angkat saudara much imam ibnu subroto kaprod mte
D4	henti kepala upt sistem kendali mutu internal spm sk rektor henti hormat saudara ahmad salim kepala upt spm hitung tanggal mei

D5	henti hormat capai batas usia purna tugas tenaga kajar fakultas dokter unissula sk ybwsa erang henti hormat capai batas usia purna tugas tenaga kajar fakultas dokter unissula dr hj utar hitung tanggal mei
D6	henti hormat capai batas usia purna tugas tenaga kajar fakultas dokter unissula sk ybwsa erang henti hormat capai batas usia purna tugas tenaga kajar fakultas dokter unissula dr h muktasim bil sps hitung tanggal februari

Selanjutnya, proses yang dilakukan yaitu Indeksasi/Tokenisasi dan Hitung TF (*Term Frequency*), DF (*Document Frequency*), dan IDF (*Invers Document Frequency*)
1.Pisahkan tiap kata
2.Kata yang sama dianggap menjadi 1 kata
3.TF : Hitung jumlah kata yang muncul pada Dokumen
4.DF : Jumlah Seluruh TF
5.IDF : Invert DF

Tabel 3.3 Hasil tokenisasi

Kata Kunci	Q	D1	D2	D3	D4	D5	D6	df	idf = log(n/df)
achmad	0	2	0	0	0	0	0	2	0,54406804
ahmad	0	0	0	0	1	0	0	1	0,84509804
anggota	0	0	1	0	0	0	0	1	0,84509804
angkat	0	0	0	1	0	0	0	1	0,84509804
batas	0	2	0	0	0	2	2	6	0,06694679
bil	0	0	0	0	0	0	1	1	0,84509804
capai	0	2	0	0	0	2	2	6	0,06694679
choliq	0	0	1	0	0	0	0	1	0,84509804
dahl	0	0	1	0	0	0	0	1	0,84509804
dasar	0	0	1	0	0	0	0	1	0,84509804
dokter	0	0	0	0	0	2	2	4	0,24303805
dr	0	0	1	0	0	1	1	3	0,36797679
elektro	0	0	0	1	0	0	0	1	0,84509804
erang	0	1	0	0	0	1	1	3	0,36797679
fakultas	0	0	0	0	0	2	2	4	0,24303805
farouq	0	2	0	0	0	0	0	2	0,54406804
februari	0	0	0	0	0	0	1	1	0,84509804
h	0	0	0	0	0	0	1	1	0,84509804
henti	1	2	1	1	2	2	2	11	-0,19629465
hitung	0	1	0	0	1	1	1	4	0,24303805
hj	0	0	0	0	0	1	0	1	0,84509804
hormat	0	2	0	0	1	2	2	7	0
ibnu	0	0	1	1	0	0	0	2	0,54406804
imam	0	0	0	1	0	0	0	1	0,84509804
internal	0	0	0	0	1	0	0	1	0,84509804

ISSN: 1693-3656

Kata Kunci	Q	D1	D2	D3	D4	D5	D6	df	idf = log(n/df)
kajar	0	0	0	0	0	2	2	4	0,24303805
kangkat	0	0	1	1	0	0	0	2	0,54406804
kaprod	0	0	0	1	0	0	0	1	0,84509804
kendali	0	0	0	0	1	0	0	1	0,84509804
kepala	0	0	0	0	2	0	0	2	0,54406804
ketua	0	0	0	1	0	0	0	1	0,84509804
khajar	0	0	1	0	0	0	0	1	0,84509804
magister	0	0	0	1	0	0	0	1	0,84509804
mei	0	1	0	0	1	1	0	3	0,36797679
mte	0	0	0	1	0	0	0	1	0,84509804
much	0	0	0	1	0	0	0	1	0,84509804
muktasim	0	0	0	0	0	0	1	1	0,84509804
mutu	0	0	0	0	1	0	0	1	0,84509804
prof	0	0	2	0	0	0	0	2	0,54406804
program	0	0	0	1	0	0	0	1	0,84509804
purna	0	2	0	0	0	2	2	6	0,06694679
rektor	0	0	0	0	1	0	0	1	0,84509804
salim	0	0	0	0	1	0	0	1	0,84509804
saudara	0	0	0	1	1	0	0	2	0,54406804
sdr	0	2	0	0	0	0	0	2	0,54406804
senat	0	0	1	0	0	0	0	1	0,84509804
sistem	0	0	0	0	1	0	0	1	0,84509804
sk	0	0	1	0	1	1	1	4	0,24303805
spm	0	0	0	0	2	0	0	2	0,54406804
sps	0	0	0	0	0	0	1	1	0,84509804
subroto	0	0	0	1	0	0	0	1	0,84509804
surat	0	1	0	0	0	0	0	1	0,84509804
tambah	0	0	1	0	0	0	0	1	0,84509804
tanggal	0	1	0	0	1	1	1	4	0,24303805
teknik	0	0	0	1	0	0	0	1	0,84509804
tenaga	0	0	0	0	0	2	2	4	0,24303805
tugas	0	2	0	0	0	2	2	6	0,06694679
undur	0	0	1	0	0	0	0	1	0,84509804
unissula	0	0	1	0	0	2	2	5	0,14612804
upt	0	0	0	0	2	0	0	2	0,54406804
usia	0	2	0	0	0	2	2	6	0,06694679
utar	0	0	0	0	0	1	0	1	0,84509804
widodo	0	0	1	0	0	0	0	1	0,84509804
ybwsa	0	1	0	0	0	1	1	3	0,36797679

Setelah ditemukan nilai TF (*Term Frequency*), DF (*Document Frequency*), dan IDF (*Invers Document Frequency*) pada setiap term, maka selanjutnya dilakukan perhitungan bobot wdt (*weight document term*) dari setiap term yang ada.

Tabel 3.4 Tabel hasil perhitungan wdt (*weight document term*).

Kata Kunci	Q	D1	D2	D3	D4	D5	D6
achmad	0	1,08813609	0	0	0	0	0
ahmad	0	0	0	0	0,845098	0	0
anggota	0	0	0,845098	0	0	0	0
angkat	0	0	0	0,845098	0	0	0
batas	0	0,13389358	0	0	0	0,133894	0,133894
bil	0	0	0	0	0	0	0,845098
capai	0	0,13389358	0	0	0	0,133894	0,133894
choliq	0	0	0,845098	0	0	0	0
dahl	0	0	0,845098	0	0	0	0
dasar	0	0	0,845098	0	0	0	0
dokter	0	0	0	0	0	0,486076	0,486076
dr	0	0	0,367977	0	0	0,367977	0,367977
elektro	0	0	0	0,845098	0	0	0
erang	0	0,36797679	0	0	0	0,367977	0,367977
fakultas	0	0	0	0	0	0,486076	0,486076
farouq	0	1,08813609	0	0	0	0	0
februari	0	0	0	0	0	0	0,845098
h	0	0	0	0	0	0	0,845098
henti	-0,196294645	-0,39258929	-0,19629	-0,19629	-0,39259	-0,39259	-0,39259
hitung	0	0,24303805	0	0	0,243038	0,243038	0,243038
hj	0	0	0	0	0	0,845098	0
hormat	0	0	0	0	0	0	0
ibnu	0	0	0,544068	0,544068	0	0	0
imam	0	0	0	0,845098	0	0	0
internal	0	0	0	0	0,845098	0	0
kajar	0	0	0	0	0	0,486076	0,486076
kangkat	0	0	0,544068	0,544068	0	0	0
kaprod	0	0	0	0,845098	0	0	0
kendali	0	0	0	0	0,845098	0	0
kepala	0	0	0	0	1,088136	0	0
ketua	0	0	0	0,845098	0	0	0
khajar	0	0	0,845098	0	0	0	0
magister	0	0	0	0,845098	0	0	0
mei	0	0,36797679	0	0	0,367977	0,367977	0

Kata Kunci	Q	D1	D2	D3	D4	D5	D6
mte	0	0	0	0,845098	0	0	0
much	0	0	0	0,845098	0	0	0
muktasim	0	0	0	0	0	0	0,845098
mutu	0	0	0	0	0,845098	0	0
prof	0	0	1,088136	0	0	0	0
program	0	0	0	0,845098	0	0	0
purna	0	0,13389358	0	0	0	0,133894	0,133894
rektor	0	0	0	0	0,845098	0	0
salim	0	0	0	0	0,845098	0	0
saudara	0	0	0	0,544068	0,544068	0	0
sdr	0	1,08813609	0	0	0	0	0
senat	0	0	0,845098	0	0	0	0
sistem	0	0	0	0	0,845098	0	0
sk	0	0	0,243038	0	0,243038	0,243038	0,243038
spm	0	0	0	0	1,088136	0	0
sps	0	0	0	0	0	0	0,845098
subroto	0	0	0	0,845098	0	0	0
surat	0	0,84509804	0	0	0	0	0
tambah	0	0	0,845098	0	0	0	0
tanggal	0	0,24303805	0	0	0,243038	0,243038	0,243038
teknik	0	0	0	0,845098	0	0	0
tenaga	0	0	0	0	0	0,486076	0,486076
tugas	0	0,13389358	0	0	0	0,133894	0,133894
undur	0	0	0,845098	0	0	0	0
unissula	0	0	0,146128	0	0	0,292256	0,292256
upt	0	0	0	0	1,088136	0	0
usia	0	0,13389358	0	0	0	0,133894	0,133894
utar	0	0	0	0	0	0,845098	0
widodo	0	0	0,845098	0	0	0	0
ybwsa	0	0,36797679	0	0	0	0,367977	0,367977

Setelah dilakukan perhitungan wdt (*weight document term*) dengan hasil masing masing term dokumen memiliki bobot tersendiri, selanjutnya perlu dilakukan perhitungan perkalian vektor antara Vektor Q dan Vektor D pada tiap term.

Tabel 3.5 Perkalian vektor Q dan D

Kata Kunci	Q	D1	D2	D3	D4	D5	D6
achmad	0	0	0	0	0	0	0
ahmad	0	0	0	0	0	0	0
anggota	0	0	0	0	0	0	0
angkat	0	0	0	0	0	0	0
batas	0	0	0	0	0	0	0

Kata Kunci	Q	D1	D2	D3	D4	D5	D6
bil	0	0	0	0	0	0	0
capai	0	0	0	0	0	0	0
choliq	0	0	0	0	0	0	0
dahl	0	0	0	0	0	0	0
dasar	0	0	0	0	0	0	0
dokter	0	0	0	0	0	0	0
dr	0	0	0	0	0	0	0
elektro	0	0	0	0	0	0	0
erang	0	0	0	0	0	0	0
fakultas	0	0	0	0	0	0	0
farouq	0	0	0	0	0	0	0
februari	0	0	0	0	0	0	0
h	0	0	0	0	0	0	0
henti	0,038531588	0,07706318	0,038532	0,038532	0,077063	0,077063	0,077063
hitung	0	0	0	0	0	0	0
hj	0	0	0	0	0	0	0
hormat	0	0	0	0	0	0	0
ibnu	0	0	0	0	0	0	0
imam	0	0	0	0	0	0	0
internal	0	0	0	0	0	0	0
kajar	0	0	0	0	0	0	0
kangkat	0	0	0	0	0	0	0
kaprod	0	0	0	0	0	0	0
kendali	0	0	0	0	0	0	0
kepala	0	0	0	0	0	0	0
ketua	0	0	0	0	0	0	0
khajar	0	0	0	0	0	0	0
magister	0	0	0	0	0	0	0
mei	0	0	0	0	0	0	0
mte	0	0	0	0	0	0	0
much	0	0	0	0	0	0	0
muktasim	0	0	0	0	0	0	0
mutu	0	0	0	0	0	0	0
prof	0	0	0	0	0	0	0
program	0	0	0	0	0	0	0
purna	0	0	0	0	0	0	0
rektor	0	0	0	0	0	0	0
salim	0	0	0	0	0	0	0
saudara	0	0	0	0	0	0	0

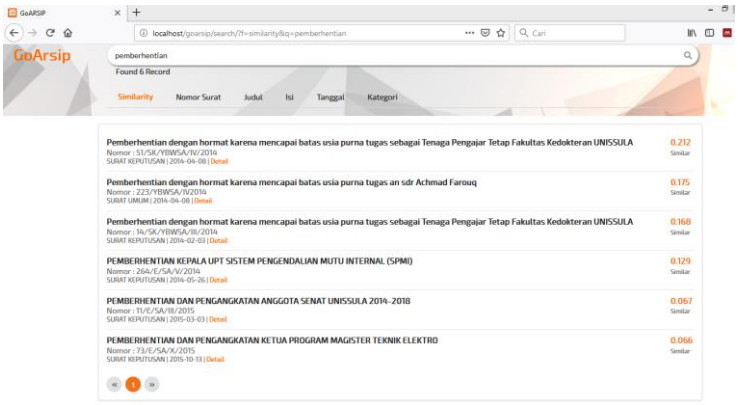
Kata Kunci	Q	D1	D2	D3	D4	D5	D6
sdr	0	0	0	0	0	0	0
senat	0	0	0	0	0	0	0
sistem	0	0	0	0	0	0	0
sk	0	0	0	0	0	0	0
spm	0	0	0	0	0	0	0
sps	0	0	0	0	0	0	0
subroto	0	0	0	0	0	0	0
surat	0	0	0	0	0	0	0
tambah	0	0	0	0	0	0	0
tanggal	0	0	0	0	0	0	0
teknik	0	0	0	0	0	0	0
tenaga	0	0	0	0	0	0	0
tugas	0	0	0	0	0	0	0
undur	0	0	0	0	0	0	0
unissula	0	0	0	0	0	0	0
upt	0	0	0	0	0	0	0
usia	0	0	0	0	0	0	0
utar	0	0	0	0	0	0	0
widodo	0	0	0	0	0	0	0
ybwsa	0	0	0	0	0	0	0
Total D0.Di	0,038531588	0,07706318	0,038532	0,038532	0,077063	0,077063	0,077063

setelah itu ditentukan tingkat kemiripan dokumen yang ditunjukkan pada tabel 3.6.

Tabel 3.6 Hasil perhitungan cosine similarity.

Proses	Q	D1	D2	D3	D4	D5	D6
Q.D	0,038531588	0,07706318	0,038532	0,038532	0,077063	0,077063	0,077063
|V|	0,196294645	2,2437537	2,908287	2,963555	3,051918	1,849721	2,329943
|Q|.|V|		0,44043684	0,570881	0,58173	0,599075	0,36309	0,457355
Similarity		0,175	0,067	0,066	0,129	0,212	0,168

Kata kunci dari data masukan dicocokkan ke dalam dokumen yang ada dalam database. Contoh pencarian didalam sistem yang sudah diterapkan metode *cosine similarity* dengan menggunakan kata kunci “pemberhentian”, akan muncul daftar dokumen dengan urutan dari tingkat *similarity* tertinggi hingga terendah ditunjukkan pada gambar 3.1.



Gambar 3.1 Hasil pencarian di sistem

Dalam implementasi algoritma *cosine similarity* tingkat keberhasilan diukur dari seberapa besar kinerja algoritma tersebut bisa memecahkan masalah yang ada, metode *cosine similarity* diuji dengan menggunakan pengukuran *precision* (presisi) dan pengukuran *recall*. Perhitungan *precision* dan *recall* ditunjukkan pada tabel 3.7.

Tabel 3.7. Tabel Hasil Pengujian *precision* dan *recall*

	Dokumen Relevan	Dokumen Tidak Relevan	Total
Dokumen yang ditemukan	**a (*hits*)** 16	**b (*noise*)** 2	**a+b** **18**
Dokumen yang tidak ditemukan	**c (*misses*)** 5	**d (*reject*)** 0	**c+d** **3**
Total	**a+c** **21**	**b+d** **0**	**a+b+c+d** **21**

Keterangan:
a (*hits*) : Dokumen yang relevan
b (*noise*) : Dokumen yang tidak relevan
c (*misses*) : Dokumen relevan yang tidak ditemukan
d (*reject*) : Dokumen tidak relevan yang tidak ditemukan
maka untuk menghitung *precision* dengan hasil sesuai Tabel 6 sebagai berikut:

$$precision = \frac{Jumlah\ dokumen\ relevan\ yang\ terpanggil\ (a)}{Jumlah\ dokumen\ terpanggil\ dalam\ pencarian\ (a+b)} x100$$

$$precision = \frac{16}{18} x100\ = 88.8\%$$

dan untuk menghitung *recall* dengan hasil sesuai Tabel 6 sebagai berikut:

$$Recall = \frac{Jumlah\ dokumen\ relevan\ yang\ terpanggil\ (a)}{Jumlah\ dokumen\ relevan\ yang\ ada\ di\ database\ (a+c)} x100$$

$$Recall = \frac{16}{21} x100 = 76.1\%$$

4. Conclusion

Algoritma *Cosine Similarity* berhasil melakukan pencarian dokumen yang mirip atau yang relevan dengan memasukkan kata kunci dalam pencarian dokumen. Algoritma *Cosine Similarity* mampu menemukan dokumen dengan tingkat kemiripan yang tinggi sehingga dapat menemukan dokumen yang relevan, hal ini ditunjukkan dengan pengukuran kinerja Algoritma *Cosine Similarity* menunjukkan angka *precision* 88.8% dan *recall* 76.1%.

References

[1] R. A. Pascapraharastyan, A. Supriyanto, and P. Sudarmaningtyas, "Rancang Bangun Sistem Informasi Manajemen Arsip Rumah Sakit Bedah Surabaya Berbasis Web," *Sist. Inf.*, vol. 3, no. 1, pp. 72–77, 2014.

[2] M. Rifauddin, "Pengelolaan Arsip Elektronik Berbasis Teknologi," *Khizanah Al- Hikmah J. Ilmu Perpustakaan, Informasi, dan Kearsipan*, vol. 4, no. 2, pp. 168–178, 2016.

[3] O. Nurdiana, J. Jumadi, and D. Nursantika, "Perbandingan Metode Cosine Similarity Dengan Metode Jaccard Similarity Pada Aplikasi Pencarian Terjemah Al-Qur'an Dalam Bahasa Indonesia," *J. Online Inform.*, vol. 1, no. 1, p. 59, 2016.

[4] S. Shum, N. Dehak, R. Dehak, and J. R. Glass, "Unsupervised Speaker Adaptation based on the Cosine Similarity for Text-Independent Speaker Verification," *Proc. Odyssey*, 2010.

[5] V. Thada and V. Jaglan, "Comparison of jaccard, dice, cosine similarity coefficient to find best fitness value for web retrieved documents using genetic algorithm," *Int. J. Innov. Eng. Technol.*, vol. 2, no. 4, pp. 202–205, 2013.

[6] R. Mihalcea, C. Corley, and C. Strapparava, "Corpus-based and knowledge-based measures of text semantic similarity," *Proc. Natl. Conf. Artif. Intell.*, vol. 1, pp. 775–780, 2006.

[7] M. E. Scholar, N. Engineering, T. Nadu, and T. Nadu, "a S Urvey on S Imilarity M Easures in T Ext M Ining," vol. 3, no. 1, pp. 19–28, 2016.

[8] J. Ramos, "Using TF-IDF to Determine Word Relevance in Document Queries," *New Educ. Rev.*, vol. 42, no. 4, pp. 40–51, 2003.

[9] B. Li and L. Han, "Distance weighted cosine similarity measure for text classification," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8206 LNCS, pp. 611–618, 2013.

[10] B. Trstenjak, S. Mikac, and D. Donko, "KNN with TF-IDF based framework for text categorization," *Procedia Eng.*, vol. 69, pp. 1356–1364, 2014.

[11] G. Sidorov, A. Gelbukh, H. Gómez-Adorno, and D. Pinto, "Soft similarity and soft cosine measure: Similarity of features in vector space model," *Comput. y Sist.*, vol. 18, no. 3, pp. 491–504, 2014.

[12] Novianto, A., 2019 "Sistem Informasi Kearsipan Menggunakan Algoritma Cosine Similarity" Pada Biro Administrasi Umum Universitas Islam Sultan Agung, Semarang