

Implentasi Information Retrieval System Menggunakan Metode Latent Semantic

Implementation Of Information Retrieval System Using The Latent Semantic Method

Microsoft Aditya Nugraha Kalimantan^a, Fhery Agustin^b

^aInformatika, Universitas Potensi Utama, Medan

^b Informatika, Universitas Potensi Utama, Medan
sendalkucing086@gmail.com¹

ABSTRAK

Pada studi ini digunakan tata metode Latent Semantic Indexing(LSI). LSI mengenakan tata cara Singular Value Decomposition(SVD) buat mereduksi dimensi. Pada studi ini digunakan 37 pasal, dan 4 kasus maupun deskripsi pelanggaran sebagai query. Sistem menerima masukkan berupa query maupun deskripsi kasus pelanggaran sehabis itu menghitung dan membenarkan pasal yang terpaut. Tingkatan keberhasilan dari tata metode ini buat menghasilkan hasil pencarian yang relevan dapat dilihat melalui besar 100% buat recall, 70% buat precision dan 82% buat f- measure.

Kata Kunci: latent semantic indexing, sistem temu balik informasi, cosine similarity, tf- idf

ABSTRACT

In this research, the Latent Semantic Indexing (LSI) procedure was used. LSI uses the Singular Value Decomposition (SVD) method to reduce size. In this research, 37 articles were used, and 4 problems or descriptions of violations were used as queries. The system receives input in the form of a query or a description of the violation problem, then calculates and determines the relevant articles. The success rate of this method for creating relevant search results can be seen by 100% for recall, 70% for precision and 82% for f-measures.

Keywords: latent semantic indexing, data retrieval system, cosine similarity, tf- idf

Info Artikel :

Disubmit: 12 Februari 2023

Direview: 3 Maret 2023

Diterima : 10 April 2023

Copyright © 2023 - Journal UPU. All rights reserved.

1. PENDAHULUAN

Bersamaan dengan cepatnya perkembangan data digital, menyebabkan terus menjadi besarnya ukuran informasi ataupun data yang terdapat. Ukuran informasi yang besar bisa merangsang data yang berarti menyebar sangat luas, sehingga jadi kurang informatif. Perihal tersebut pula menyebabkan berkembangnya teknologi data yang diperlukan oleh pengguna sehingga menyebabkan timbulnya sesuatu cabang ilmu baru dalam teknologi data ialah information retrieval(IR)(T. Wicaksono, 2005). Cabang ilmu tersebut pasti dibutuhkan supaya memudahkan dalam melaksanakan pencarian informasi berukuran yang besar tersebut.

Pelaksanaan information retrieval(IR) yang kerap ditemukan merupakan pada mesin pencari(search engine). Mesin pencari(search engine) ialah kombinasi fitur keras dan fitur lunak komputer yang disediakan oleh industri tertentu melalui website yang telah ditetapkan (Abdillah, Falkner, & Hemer, 2010).

Salah satu pemanfaatan search engine ini merupakan dalam bidang kesehatan. Dimana digunakan buat melaksanakan pencarian terhadap bermacam keluhan kesehatan yang dirasakan para pengguna internet. Tetapi perkaranya merupakan konsumsi mesin pencari masih kerap mempersulit pengguna, antara lain pada dikala mengumpulkan E- Boo, hasil percarian sangat luas sehingga data yang dicari kurang khusus. Disamping itu pula, banyak data yang hasilkan dari web yang masih diragukan kevalidannya sebab tidak tertera pakar penanggungjawabnya. Dalam riset ini, hendak diusulkan sesuatu sistem information retrieval(IR) yang khusus buat mempermudah users dalam melaksanakan pencarian

tentang permasalahan kesehatan. Informasi yang digunakan diambil dari sebagian web kesehatan yang diseleksi.

Secara garis besar, pokok ulasan IR meliputi 2 perihal ialah pembobotan serta perangkingan(Holle, Arifin,& Purwitasari, 2015). Perangkingan digunakan buat memperoleh dokumen- dokumen yang relevan dengan query pengguna diurutkan dari tingkatan relevansinya. Perangkingan ini dibutuhkan sebab bisa jadi dokumen yang cocok dengan query pencarian lebih dari satu.

Salah satu tata metode yang dapat digunakan dalam perangkingan dokumen ialah Vector Ruang Model(VSM). Pada tata metode ini dokumen direpresentasikan sebagai sesuatu vektor yang dibentuk dari nilai- nilai term yang jadi indeksinya. Harrag dkk(2008) mengenakan vector ruang model berbasis term weighting TF- IDF buat melakukan perangkingan pada dokumen berbahasa Arab. TF. IDF yakni kombinasi dari tata metode Term Frequency(TF) yang yakni mengukur kepadatan term dalam dokumen, dengan tata metode Inverse Term Frequency(IDF) yang mengukur kepentingan sesuatu term.

Buat tingkatan relevansi query terhadap hasil pencarian, diperlukan tata cara buat memikirkan ikatan semantic antar term. Hingga digunakan tata metode Latent Semantic Indexing(LSI) yang mampu mengidentifikasi makna semantik antar term bersumber pada pola jalinan yang ada pada koleksi dokumen/ artikel. Sifat utama LSI ialah kemampuannya buat membangun jalinan antara istilah yang mencuat dalam konteks yang seragam(Wahib, Pasnur, Santika,& Arifin, 2015). Tidak hanya itu, LSI dibutuhkan sebab dalam ulasan kesehatan, banyak sekali istilah- istilah yang dapat mempunyai arti berbeda- beda, bergantung pada pelaksanaan konteksnya.

Oleh sebab itu, dalam riset ini diusulkan campuran tata cara TF. IDF dengan tata cara Latent Semantic Indexing(LSI) yang diimplementasikan pada sistem information retrieval(IR). Pemakaian kedua tata cara TF. IDF serta LSI ini diharapkan bisa tingkatan relevansi perangkingan postingan E- book yang pas serta cocok yang diharapkan users. Dengan latar balik yang terdapat hingga Fitur Lunak yang dibentuk bertajuk“ **IMPLENTASI INFORMATION RETRIEVAL SYSTEM Memakai Tata cara LATENT SEMANTIC**”.

2. METODE

Pada sesi ini dicoba dengan menekuni teori bawah yang menunjang riset, pencarian serta pengumpulan data- data yang diperlukan. Buat mengumpulkan informasi yang diperlukan, hingga penulis mengenakan metode selaku berikut:

a) Target (*OBSERVASI*)

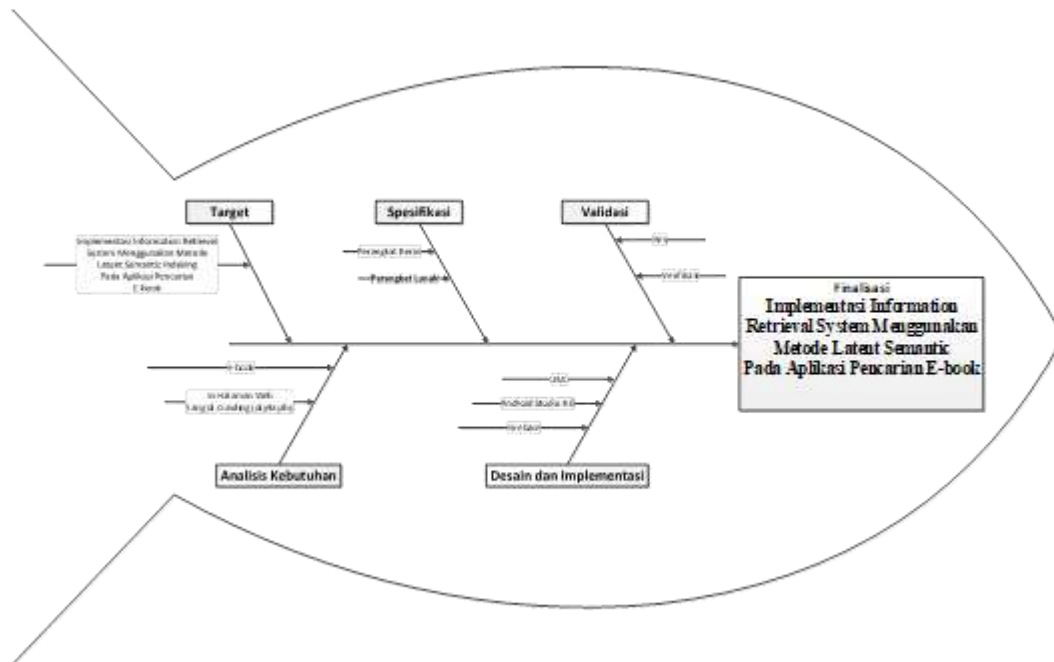
Adapun target dari penelitian ini adalah dapat membangun suatu Implementasi Information Retrieval System Memakai Tata cara Latent Semantic Indexing Pada Aplikasi Pencarian E- book

b) Analisis Kebutuhan

Kebutuhan hendak sesuatu data menuntut seorang buat menelusuri segala koleksi dokumen yang dimilikinya. Banyaknya dokumen yang tersimpan di dalam pc hendak membuat pengguna hadapi kesusahan buat mendapatkan data yang diperlukan. Pengguna tidak bisa memandang isi dokumen satu persatu buat memperoleh data yang pas, Buat memperoleh data pengguna biasa memakai sarana pencarian dokumen yang disediakan oleh sistem pembedahan dengan memasukkan kata kunci

A. Metodologi Penelitian

Penelitian ini akan melalui beberapa tahapan. Tahapan dalam penelitian ini dapat dimodelkan pada diagram *Fishbone*. Adapun beberapa tahapan yang digunakan dalam penelitian ini adalah sebagai berikut:



Gambar 1. Diagram *Fishbone* Implementasi Information Retrieval System Mengenakan Laten Semantic Pada Aplikasi Pencarian E- Book

Keterangan :

1. Analisa Kebutuhan

Pada tahapan ini ialah analisa terhadap kebutuhan yang dibutuhkan buat menggapai tujuan riset yang hendak dicoba ialah informasi Pengaduan Pelayanan, serta informasi User.

2. Desain Sistem

Desain sistem yang digunakan dalam teori merupakan pemodelan UML ialah Use Case Diagram, Class Diagram, Activity Diagram serta Sequence Diagram.

3. Tools

Pada tahapan ini pengamat mengenakan aplikasi Android studio, PHP, xampp dan pengamat mengenakan hardware komputer/ laptop. Database yang digunakan ialah MySQL.

4. Pengujian

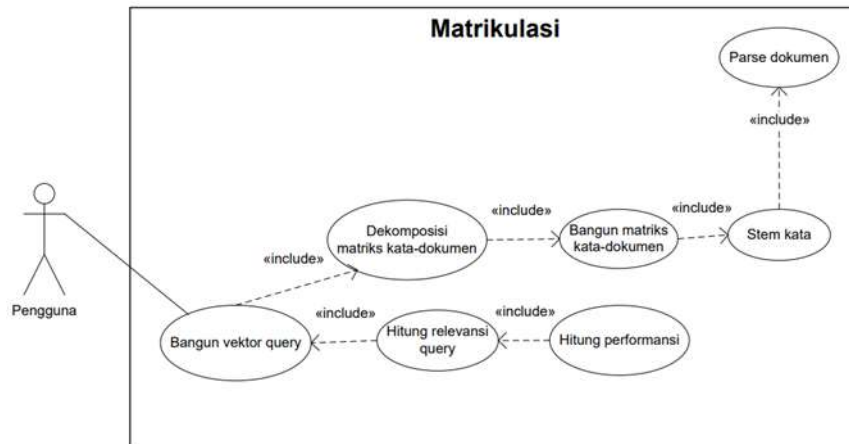
Pada tahapan ini pengamat menguji sistem yang telah dibuat mengenakan pengujian teori dan praktek. Pengujian teori pengamat mengenakan blackbox testing dan pengujian praktek pengamat mengenakan localhost.

5. Hasil

Pada tahapan ini riset telah berakhir terbuat. Hasil dari riset ini ialah aplikasi pengaduan pelayanan warga pada Komplek Ray Pendopo 4. Medan berbasis website serta Android.

3. HASIL DAN PEMBAHASAN

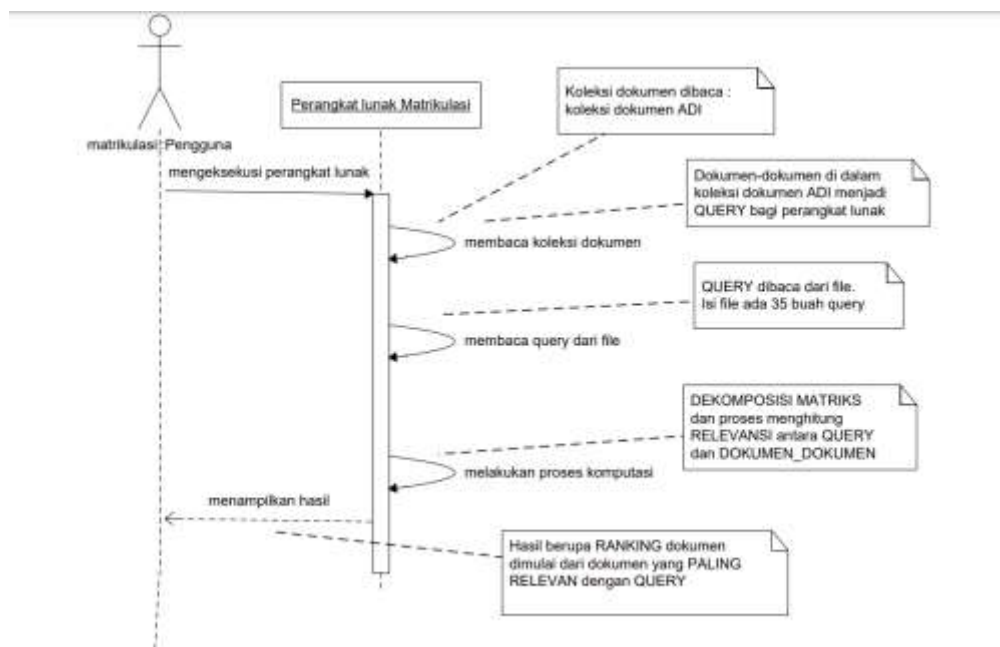
Dalam penataan sesuatu program dibutuhkan sesuatu model informasi yang berupa diagram yang bisa menarangkan sesuatu alur proses sistem yang hendak di bangun. Dalam penyusunan skripsi ini penulis memakai tata cara UML yang dalam tata cara itu penulis mempraktikkan diagram Use Case. Hingga digambarlah sesuatu wujud diagram Use Case yang bisa dilihat pada foto 2 berikut :



Gambar 2. Use Case Diagram Rancang Bangun Retrieval Information

Rancangan kelas-kelas yang akan digunakan pada sistem yang akan dirancang dapat dilihat pada gambar dibawah ini :

Pendefinisian Skenario Penggunaan Perangkat Lunak

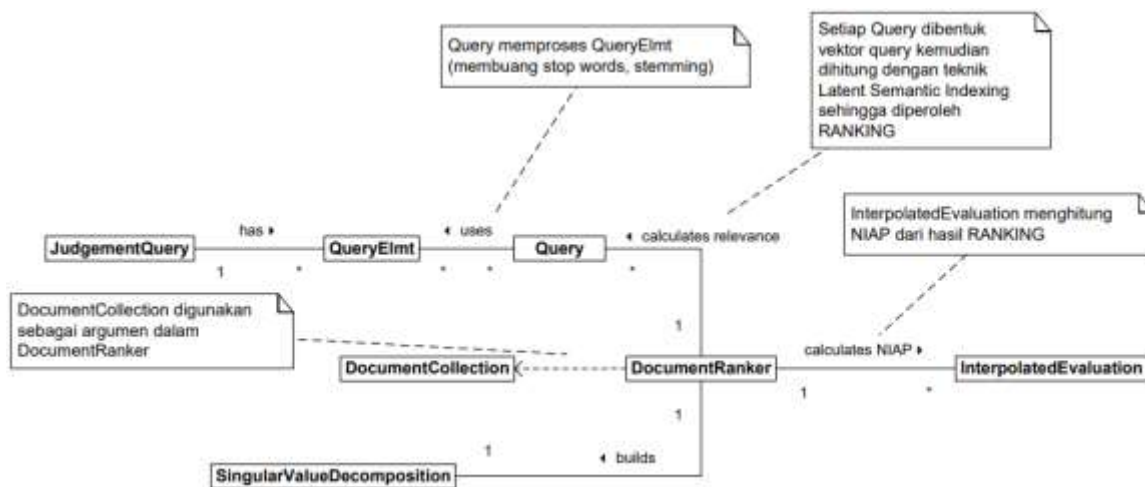


Gambar 3. Squence Diagram dari Squence Diagram(High Tingkat)

Urutan proses dalam Gambar diatas :

- Pengguna mengeksekusi fitur lunak(PL) Matriulasi.
- PL membaca koleksi dokumen.
3. PL membaca query dari file. Dalam file ada 35 buah query. Kelompok query ini informasi uji untuk PL.

- d) PL melaksanakan proses komputasi buat menghitung relevansi query dengan dokumen- dokumen.
 - e) PL menampilkan hasil berupa ranking dokumen- dokumen, dimulai dari dokumen yang sangat relevan dengan query. Catatan ranking dokumen ada 35 buah. Tiap- masing- masing yakni catatan ranking buat masing- masing query. Tidak cuma ranking dokumen, pula ditampilkan nilai NIAP dari masing- masing query.
- Pendefinisian Class Diagram Tahap Awal



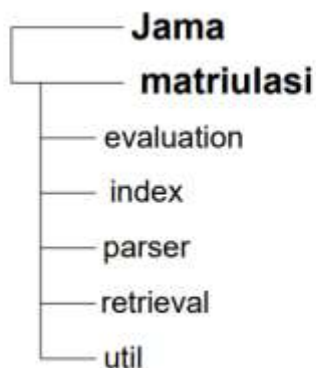
Gambar 4. Class Diagram Sesi Dini

Kelas- kelas dalam class diagram dini ditafsirkan pada foto diatas meliputi:

1. Kelas JudgementQuery
2. Kelas QueryElmt
3. Kelas Query
4. Kelas DocumentCollection
5. Kelas DocumentRanker
6. Kelas SingularValueDecomposition
7. Kelas InterpolatedEvaluation

Perancangan Package Kode Program Matriulasi

Kode program Matriulasi dibuat dalam bahasa pemrograman Java™ dan mengenakan kompiler Java™, diucap Java Pengembang Kit(JDK®) jenis 1. 4. 1. Kode program dipecah dalam 2(2) package utama, yakni package Jama dan package matriulasi.



Gambar 5. Struktur Package PL Matriulasi

Package Jama

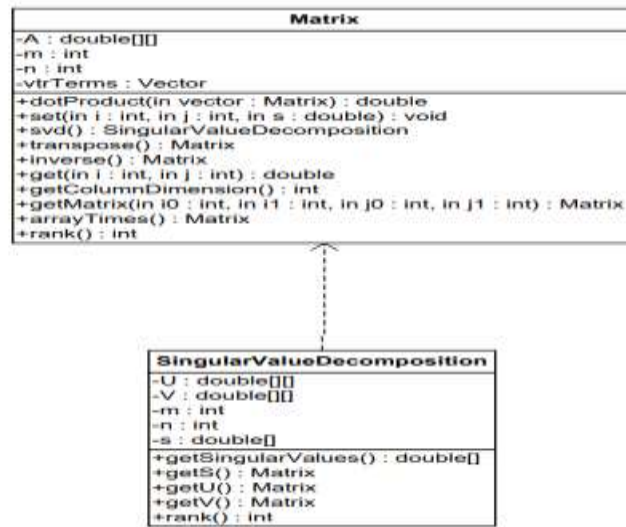
Jama yakni singkatan dari Java Matrix®, berisi kelas- kelas yang berfungsi buat melakukan operasi matriks semacam membuat struktur data matriks dan mendekomposisi suatu matriks(singular value decomposition).

Package Matriulasi

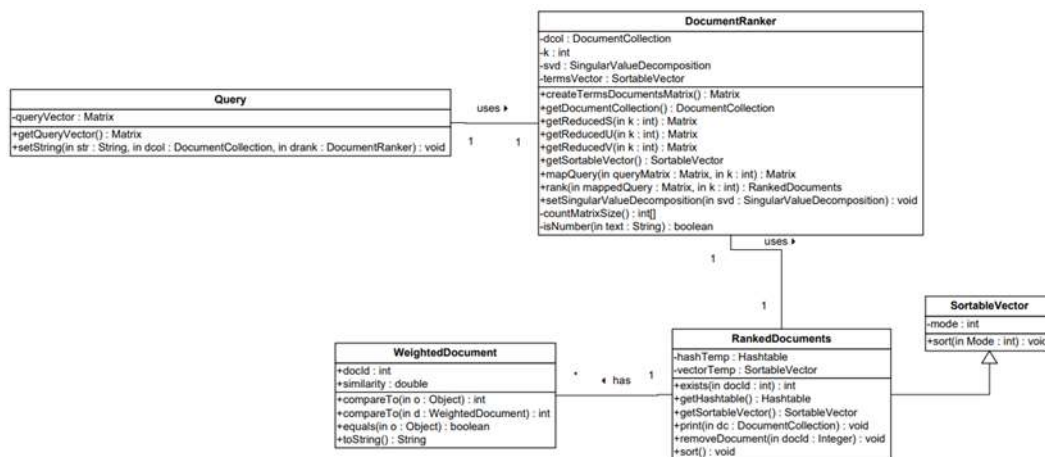
Package ini yakni inti(core) dari program Matriulasi. Package matriulasi dipecah jadi 5(5) buah sub- package, yakni evaluation, index, parser, retrieval, dan util. Penjelasan mengenai tiap- masing- masing sub- package merupakan

- a. evaluation berisi kelas- kelas yang berfungsi menghitung performansi hasil ranking
- b. index berisi kelas- kelas yang berfungsi melakukan proses indexing terhadap koleksi dokumen;
- c. parser berisi kelas- kelas yang berfungsi mengerjakan stemming terhadap hasil indexing;
- d. retrieval berisi kelas- kelas yang berfungsi melakukan komputasi buat memperoleh nilai relevansi query dengan dokumen.
- e. util berisi kelas- kelas yang berfungsi sebagai utility dalam mengerjakan proses- proses.

Class Diagram Package Jama



Gambar 6. Diagram Package Jama



Gambar 7. Class Diagram Rancang Bangun Aplikasi Pengaduan Pelayanan Masyarakat

Adapun tampilan hasil program adalah sebagai berikut :

1. Tampilan Program

Terdapat beberapa tampilan program Case Folding, Tokenizing, Stopword Removal.

```

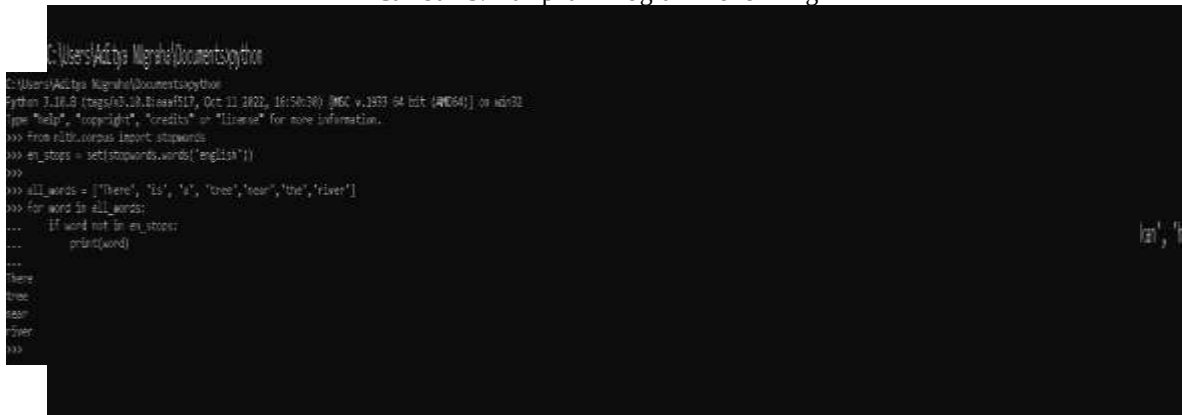
Microsoft Windows [version 10.0.22000.1098]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Aditya M\graha>python
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:58:36) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> kalimat = "Setiap orang yang pada waktu pemungutan suara dengan sengaja melakukan perbuatan melawan hukum mengaku dirinya sebagai orang lain untuk menggunakan hak pilih"
>>> CASE_FOLDING = kalimat.lower()
>>> print(CASE_FOLDING)
setiap orang yang pada waktu pemungutan suara dengan sengaja melakukan perbuatan melawan hukum mengaku dirinya sebagai orang lain untuk menggunakan hak pilih
>>>
    
```

Gambar 8. Tampilan Program Case Folding

Case folding ialah salah satu bentuk text preprocessing yang sangat sederhana dan efektif meski sering diabaikan. Tujuan dari case folding buat mengubah segala huruf dalam dokumen jadi huruf kecil. Hanya huruf 'a' sampai 'z' yang diterima. Karakter tidak cuma huruf dihilangkan dan dikira delimiter. Pada tahap ini tidak mengenakan external library apapun, kita bisa memakai modul yang terdapat di python.

Gambar 9. Tampilan Program Tokenizing



```

C:\Users\Aditya Nugraha\Documents>python
Python 3.10.8 (tags/v3.10.8:10f133f, Oct 11 2022, 16:50:26) [AMD64] on win32
Type "help", "copyright", "credits" or "license()" for more
>>> from re import split
>>> words = ["There", "is", "a", "tree", "near", "the", "river"]
>>> for word in words:
...     if word not in words:
...         print(word)
...
There
is
a
tree
near
the
river
>>>

```

Gambar 10. Tampilan Program Stopword Removal

2. Sesi text preprocessing

- Case Folding

Pada sesi ini seluruh kepribadian huruf diganti jadi huruf kecil.

Tabel 1. Case Folding

Sebelum CaseFolding	Sesudah CaseFolding
Masing- masing orang yang pada waktu pemungutan suara dengan terencana melakukan perbuatan melawan hukum mengaku dirinya sebagai orang lain buat mengenakan hak pilih	Masing- masing orang yang pada waktu pemungutan suara dengan terencana melakukan perbuatan melawan hukum mengaku dirinya sebagai orang lain buat mengenakan hak pilih

- Tokenizing

Pada tokenizing dicoba penghilangan karakteristik baca dan pemotongan kata pada kalimat dalam dokumen

Tabel 2. Tokenizing

Sebelum Tokenizing	Sesudah Tokenizing
Masing- masing orang yang pada waktu pemungutan suara dengan terencana melakukan perbuatan melawan hukum mengaku dirinya sebagai orang lain buat mengenakan hak pilih	[" masing- masing", " orang", " yang", " pada", " waktu", " pemungutan", " suara", " dengan", " terencana", " melakukan", " perbuatan", " melawan", " hukum", " mengaku", " dirinya", " sebagai", " orang", " lain", " buat", " mengenakan", " hak", " pilih"]

3. Stop Removal

Tahap selanjutnya ialah Stopword Removal ialah proses pembuangan stopword buat mengurangi volume kata sehingga hasil pencarian jadi lebih akurat karena noise hendak direduksi.

Tabel 3. Stop Removal

Sebelum <i>StopwordRemoval</i>	Sesudah <i>StopwordRemoval</i>
['There', 'is', 'a', 'tree','near','the','river']	There tree near river

4. Pembobotan TF- IDF

Dengan memakai persamaan(1) didapatkan nilai TF serta persamaan(2) digunakan buat memperoleh nilai IDF. Bobot merupakan hasil kali dari nilai TF serta IDF.

term	tf	idf	bobot
orang	2	1.27875	2.55751
mungut	1	1.46584	1.46584
suara	1	1.27875	1.27875
sengaja	1	1.14842	1.14842
buat	1	1.40369	1.40369
lawan	1	1.40369	1.40369
hukum	1	1.37566	1.37566
hak	1	1.67669	1.67669
pilih	1	1.37566	1.37566

Gambar 11. Hasil Pembobotan TF-IDF

5. Pemobotan LSI

Segala term hasil preprocessing dari segala pasal dibentuk jadi sesuatu matrik(Matrik M). Baris merepresentasikan segala term unik pada segala pasal dan Kolom merepresentasikan pasal. Bobot hasil pembobotan TF- IDF dimasukkan ke matrik bersumber pada term dan sebabnya. Apabila term tidak ada dalam pasal sampai diberi nilai 0 pada pasal tersebut.

	Pasal 1	Pasal 2	Pasal 3	Pasal 4
orang	2.55751	1.37566	0.00000	2.60241
sengaja	1.14842	1.30132	0.00000	0.00000
laku	0.00000	1.60211	0.00000	1.27875
buat	1.40369	0.00000	0.00000	1.27875
lawan	1.40369	1.46584	0.00000	1.27875
hukum	1.37566	1.46584	0.00000	1.27875
palsu	0.00000	0.00000	0.00000	0.00000
data	0.00000	0.00000	0.00000	0.00000
daftar	0.00000	0.00000	0.00000	0.00000
pilih	1.37566	1.14842	0.00000	0.00000
pasal	0.00000	1.14842	0.00000	0.00000
anggota	0.00000	0.00000	1.04113	0.00000
pps	0.00000	0.00000	1.04113	0.00000

Gambar 12. Ilustrasi Matrik Meter

Sehabis matriks terbentuk, selanjutnya matriks tersebut hendak di dekomposisi mengenakan SVD. Dan menghasilkan matriks U, S, V. Dalam kaitannya biar mempermudah perhitungan selanjutnya, matriks hasil SVD di lakukan pengurangan dimensi, sebanyak nilai k dengan tujuan buat memesatkan proses. Prosesnya ialah mengambil k kolom dini dari matriks U dan V. dan mengambil k baris dan kolom dini pada matriks S. Sehingga matriks hasil SVD diucap matriks U_k , S_k , V_k .

U_k		S_k	
0.0336	0.0199	0.0255	0
0.167	0.037	0	0.04
-0.0745	0.0849		
-0.0282	0.009		
0.05	-0.0609		
0.0024	0.0167		
-0.009	0.0009		
0.008	0.0197		
0.0139	-0.0031		
0.0065	0.0639		
-0.0411	0.0777		
-0.0839	-0.0508		
0.1468	0.1433		
0.002	0.0076		
0.0363	0.0872		
-0.0007	0.1178		
0.0224	-0.0138		
0.0365	-0.012		
0.0013	0.0016		
-0.0036	-0.0309		
0.0219	0.0172		
-0.0423	0.0315		
0.0298	-0.0462		
0.1027	0.2962		
0.1185	-0.3062		
-0.0002	0.0389		
0.0464	0.0175		

V_k	
0.0252	0.0328
-0.0412	-0.1143
0.0241	0.0043
-0.0264	0.0177
-0.0376	-0.0599
0.1125	0.2574

V_k					
0.0252	-0.0412	0.0241	-0.0264	-0.0376	0.1125
0.0328	-0.1143	0.0043	0.0177	-0.0599	0.2574

Gambar 13. Sampel Matrik Hasil Dekomposisi

6. Pengujian Metode

Pada studi ini digunakan data berjumlah 37 pasal dan 4 kasus kejahatan yang terpaut dengan pelanggaran yang diatur pada pasal pada UU No. 10 Tahun 2016 sebagai query, dengan mengenakan nilai k 20, 25, 30, dan 35.

No	Query	k	P	R	F- measure
1.	Q1.	20	0,3	1	0.46
		25	0,5	0,5	0.50
		30	0,5	0,5	0.50
		35	0,5	0,5	0.50
2.	Q2	20	0,1	1	0.18
		25	0,2	1	0.33
		30	0,2	1	0.33
		35	0,3	1	0.46
3.	Q3	20	0,4	1	0.57
		25	0,4	1	0.57
		30	0,4	1	0.57
		35	0,2	0,5	0.29
4.	Q4	20	0,7	0,5	0.58
		25	0,7	1	0.82
		30	0,7	1	0.82
		35	0,7	1	0.82

Gambar 14. Hasil Pengujian Precision Recall, dan F-Measure

4. KESIMPULAN

Bersumber pada perancangan, analisa, dan implementasi serta pengujian pada penerapan tata metode latent semantic indexing pada sistem temu balik informasi pasal pada undang- undang pemilu bersumber pada kasus pelanggaran pemilu, sampai didapatkan kesimpulan:

- Dengan mempraktikkan segala yang dibutuhkan buat membangun tata metode Latent Semantic Indexing(LSI), semacam text processing, TF- IDF, SVD dan Cosine Similarity dapat membangun sesuatu sistem temu kembali informasi pada undang- undang pemilu bersumber pada kasus pelanggaran pemilu
- Tingkatan keberhasilan dari sistem temu balik informasi dengan tata metode LSI ini dapat dilihat melalui besarnya nilai uji kinerja sistem yang menghasilkan nilai recall sebesar 100%, precision sebesar 70 dan f- measure sebesar 82%. Pengujian oleh pengguna menghasilkan persentase sebesar 89, 21%(sangat besar).

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Universitas Potensi Utama yang telah banyak memberikan masukan dan saran dalam penyelesaian Skripsi ini.

REFERENSI

- [1] Al- Anzi, F. S. serta AbuZeina, D.(2017). Toward an enhanced Arabic text classification using cosine similarity and Latent Semantic Indexing. Journal of King Saud University- Computer and Information Sciences, 29(2), 189–195. Tersedia pada [https:// doi. org/ 10. 1016/ j. jksuci. 2016. 04. 001](https://doi.org/10.1016/j.jksuci.2016.04.001).
- [2] Dwiyanoro, D.(2017). Sistem Temu Kembali Dengan Keyword(Deskriptif Memakai Recall Serta Precision Pada Judul, Subjek Opac Bibliotek Universitas Gadjah Mada). Khizanah al- Hikmah: Harian Ilmu Bibliotek, Data, serta Kearsipan, 5(2), 164–175. Tersedia pada [https:// doi. org/ 10. 1016/ j. jksuci. 2016. 04. 001](https://doi.org/10.1016/j.jksuci.2016.04.001).

- 24252/ kah. v5i2a4.
- [3] Froud, H., Lachkar, A. serta Ouatik, S. A.(2013). Arabic Text Summarization Based on Latent Semantic Analysis to Enhance Arabic Documents Clustering. International Journal of Informasi Mining& Knowledge Management Process, 3(1), 79–95. Ada pada [https:// doi. org/ 10. 5121/ ijdkp. 2013. 3107](https://doi.org/10.5121/ijdkp.2013.3107).
- [4] Hasan, Meter. ismail.(2018). Information Retrieval System postingan kesehatan memakai Pembobotan TF. IDF serta Latent Semantic indexing.(Doctoral dissertation, Universitas Islam Negara Maulana Malik Ibrahim).
- [5] Khosrow- Pour, Meter.(2005). Encyclopedia of Information Science and Technology(5 Volumes). Idea Group Reference.
- [6] Wahib, A. dkk.(2015). Perangkingan Dokumen Berbahasa Arab Memakai Latent Semantic Indexing. Harian Buana Informatika, 6(2), 83–92. Ada pada [https:// doi. org/ 10. 24002/ jbi. v6i2. 411](https://doi.org/10.24002/jbi.v6i2.411).
- [7] Wahyuddin.(2017). Pemanfaatan sistem temu balik data di dinas bibliotek serta kearsipan kabupaten barru skripsi.(Doctoral dissertation, Universitas Islam Negara alauddin makassar).
- [8] Yudho Baskoro, S., Ridok, A. serta Tanzil Furqon, Meter.(2015). Pencarian Pasal Pada Kitab Undang-Undang Hukum Pidana(Kuhp) Bersumber pada Permasalahan Memakai Tata cara Cosine Similarity Serta Latent Semantic Indexing(Lsi). Journal of Enviromental Engineering and Sustainable Technology, 2(2), 83–88. Tersedia pada [https:// doi. org/ 10. 21776/ ub. jeest. 2015. 002. 02. 4](https://doi.org/10.21776/ub.jeast.2015.002.02.4).