

# PENDETEKSI KESAMAAN KATA UNTUK JUDUL PENULISAN BERBAHASA INDONESIA MENGGUNAKAN ALGORITMA *STEMMING* NAZIEF-ADRIANI

Mochamad Wisuda Sardjono<sup>1)</sup>, Margi Cahyanti<sup>2)</sup>, Maulana Mujahidin<sup>3)</sup>, dan Rini Arianty<sup>4)</sup>

<sup>1,2,3,4)</sup>Jurusan Sistem Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Gunadarma  
<sup>1,2,3,4)</sup>Jalan Margonda Raya No.100, Pondok Cina, Depok  
E-mail : moch\_wisuda@staff.gunadarma.ac.id<sup>1)</sup>, margi@staff.gunadarma.ac.id<sup>2)</sup>,  
mujahidin@staff.gunadarma.ac.id<sup>3)</sup>, rini\_a@staff.gunadarma.ac.id<sup>4)</sup>

## ABSTRAK

Penelitian ini mengimplementasikan algoritma *stemming* Nazief-Adriani untuk mencari kesamaan judul penulisan berbahasa Indonesia. Dengan mencari kesamaan judul penulisan tersebut dapat membantu mahasiswa maupun dosen dalam menentukan judul penulisan yang ingin dibuat atau dapat menjadi referensi dalam pembuatan penulisan, sehingga dapat menghindari tindakan *plagiarisme* judul penulisan. Penelitian ini juga membangun aplikasi yang dapat menambahkan data judul penulisan ke dalam *database* dan mencari kesamaan judul penulisan dari judul penulisan yang terdapat dalam *database*. Tahapan yang dilakukan untuk proses mencari kesamaan adalah memecah kata pada judul penulisan untuk didapatkan kata dasar dengan menggunakan algoritma *stemming* Nazief-Adriani, kemudian dihitung *term frequency (tf)* untuk mendapatkan persentase kesamaan kata. Hasil dari pencarian kesamaan judul penulisan ini berupa kata dasar yang diambil dari setiap kata dari judul penulisan yang ingin dicari kesamaan dan persentase kesamaan dengan judul penulisan yang ada di *database*. Pengujian dilakukan pada 5 judul penulisan yang diinputkan oleh pengguna kemudian dicari kesamaannya dengan 1561 judul penulisan pada *database*.

**Kata Kunci :** *Text Mining, Stemming, Algoritma Nazief-Adriani, Plagiarisme*

## 1. PENDAHULUAN

Perkembangan data teks saat ini telah mencapai jumlah yang cukup besar. Hal tersebut disebabkan oleh berkembangnya dunia teknologi informasi yang terdiri dari data teks di dalamnya. Saat ini, berbagai macam media *online* seperti blog, situs berita *online*, dan jejaring sosial menjadi sumber data teks yang sangat potensial untuk digali lebih dalam. Namun, data berbentuk teks memiliki karakteristik yang tidak terstruktur dan sangat banyak memuat *noise*. Oleh karena itu, *text mining* memiliki peran penting dalam bidang data mining. Dengan mengaplikasikan proses-proses dalam *text mining*, maka akan diperoleh pola-pola data dan ekstraksi dari pengetahuan-pengetahuan yang potensial dari data teks (Kao, 2005).

Pemanfaatan teknologi informasi telah menjadi kebutuhan dalam era modern saat ini. Komponen yang ada di dalam dunia digital salah satunya adalah dokumen teks. Dokumen dalam bentuk digital memudahkan dalam hal penyimpanan dan mudah untuk terjadinya *plagiarism* yang selalu menjadi sorotan terutama di kalangan akademis. Tindakan plagiat yang dilakukan seseorang ini tidak mencerminkan sikap kreatif sebagai kaum intelektual. Rawannya plagiat pada dokumen digital mendorong para peneliti untuk mengembangkan piranti lunak *plagiarism checker* untuk mendeteksi plagiat dengan cara mengukur tingkat kemiripan dokumen tersebut dengan dokumen-dokumen lainnya. Hal ini

penting karena kredibilitas akademisi maupun penerbit ditentukan oleh originalitas dokumen atau artikel yang diterbitkannya (Khan, 2011).

*Information Retrieval* (IR) merupakan area studi yang mempelajari metode pencarian serta pemisahan material berupa dokumen teks dari data yang tidak terstruktur yang memenuhi kebutuhan informasi yang relevan (Manning, 2009). IR merupakan suatu pencarian informasi yang didasarkan pada suatu query yang diharapkan dapat memenuhi keinginan user dari kumpulan dokumen yang ada (Poernomo, 2015).

*Stemming* merupakan metode untuk meningkatkan performansi IR dengan cara mentransformasi kata-kata di dalam suatu dokumen teks ke dalam bentuk kata dasarnya. Dengan melakukan proses *stemming*, maka efisiensi algoritma *Information Retrieval* dapat ditingkatkan karena menghilangkan kata-kata berimbuhan yang memiliki makna morfologi berbeda-beda namun memiliki interpretasi semantik yang sama (Manning, 2009).

Untuk bahasa Indonesia, terdapat algoritma *stemming* oleh Bobby Nazief dan Mirna Adriani (Adriani, 2007). Proses *stemming* dalam bahasa Indonesia ini lebih kompleks, karena terdapat berbagai macam variasi serta kombinasi imbuhan yang harus dihapus untuk mendapatkan kata dasar. Pada penelitian mengenai *stemming*, nilai akurasi yang diperoleh pun lebih baik daripada algoritma Vega dan Porter, berkisar antara

82,84% sampai dengan 97,931% (Simarangkir, 2017). Hasil penelitian lain didapatkan dari total 2.132 kata yang di uji, algoritma Nazief-Adriani menghasilkan kata dasar yang benar sebanyak 2.031 kata, sedangkan algoritma Porter menghasilkan kata dasar yang benar sebanyak 1.687 kata. Tingkat kesalahan *stemmer* Nazief-Adriani hanya 5%, sedangkan Porter 21% (Susyanto, 2017).

Penerapan algoritma Nazief-Adriani dengan metode *Cosine Similarity* pada penelitian tentang mencari kesamaan dokumen yang menggunakan kalimat aktif dibandingkan dengan dokumen dengan kalimat pasif dihasilkan tingkat optimalisasi sebesar 93,81% dibanding dengan algoritma lain (Azhar, 2014). Hasil *stemming* pada tahapan *pre-processing* menggunakan algoritma Nazief-Adriani menghasilkan akurasi 95% untuk teks terjemahan Al Quran dan tafsir berbahasa Indonesia. Kegagalan pembentukan kata dasar pada tahap ini dikarenakan masih adanya kata dalam bahasa Arab yang tidak diterjemahkan ke bahasa Indonesia (Poernomo, 2015).

Berdasarkan uraian diatas kemampuan Algoritma Nazief-Adriani untuk memecahkan kalimat menjadi kata dasar lebih baik dibandingkan dengan algoritma lainnya. Oleh karena itu pada penelitian ini algoritma Nazief-Adriani digunakan untuk mencari kata dasar pada kalimat. Untuk mendapatkan persentase kesamaan kata penelitian ini menggunakan metode *term frequency*. Pada penelitian ini juga dikembangkan aplikasi pendeteksi kesamaan kata untuk judul penulisan berbahasa Indonesia dengan menggunakan algoritma *stemming* Nazief-Adriani.

## 2. RUANG LINGKUP

Penelitian ini membahas beberapa masalah, diantaranya :

1. Menerapkan algoritma *stemming* Nazief-Adriani untuk memecah judul menjadi kata dasar.
2. Memanfaatkan metode *term frequency* untuk mendapatkan persentase kesamaan kata.
3. Menggunakan bahasa pemrograman Visual C# dan Microsoft Access sebagai *database* untuk mengembangkan aplikasi.

## 3. BAHAN DAN METODE

Langkah-langkah mencari kesamaan kata pada judul penulisan dibahas pada bagian ini dengan *stemming* menggunakan algoritma Nazief-Adriani dan *term frequency*.

### 3.1 Text Mining

*Text Mining* dapat didefinisikan sebagai suatu proses untuk menggali informasi yang berhubungan dengan sekumpulan dokumen dengan menggunakan *tool* analisis yang merupakan komponen pada data mining. *Text mining* berbeda dengan data mining, jenis data yang digunakan adalah *semi-structure* atau *unstructure* (Ohara, 2011).

Tantangan yang dihadapi pada *text mining* adalah jumlah data yang besar, dimensi yang tinggi dan data yang berubah (Ohara, 2011).

Dalam implementasinya, *text mining* akan mencari pola-pola tertentu, mengasosiasikan satu bagian teks dengan yang lain berdasarkan aturan-aturan tertentu, sehingga dapat dilakukan analisa keterhubungan antar teks dalam sekumpulan dokumen. Tahapan-tahapan pada *text mining* (Ohara, 2011) :

1. *Case Folding*  
Mengubah semua huruf atau teks dalam dokumen menjadi huruf kecil dan menghilangkan karakter selain huruf.
2. *Tokenizing*  
Memotong tiap kata dalam kalimat dengan menggunakan spasi sebagai *delimiter* yang akan menghasilkan token berupa kata.
3. *Filtering*  
Memisahkan kata yang dianggap tidak penting atau tidak memiliki makna, hasil dari tahap sebelumnya.
4. *Stemming*  
Mengembalikan kata-kata yang diperoleh dari tahap sebelumnya ke bentuk dasarnya, menghilangkan imbuhan awal (prefix) dan imbuhan akhir (sufix) sehingga didapatkan kata dasarnya. Pada tahapan ini memerlukan masukan berupa kata yang terdapat dalam suatu dokumen dengan menghasilkan keluaran berupa *root word*.
5. *Tagging*  
Mengubah kata dalam bentuk lampau menjadi bentuk sekarang.
6. *Analyzing*  
Keterhubungan antar kata dalam dokumen akan ditentukan dengan menghitung *term frequency* pada dokumen. Tahapan ini dikenal dengan tahapan pembobotan.

### 3.2 Algoritma Stemming Nazief-Adriani

*Stemming* adalah cara yang digunakan untuk meningkatkan performa *Information Retrieval* dengan cara mentransformasikan kata-kata dalam sebuah dokumen teks ke kata dasarnya. Proses *stemming* pada teks Bahasa Indonesia digunakan untuk menghilangkan sufiks, konfiks, dan prefiks. Hal ini berbeda dengan teks Bahasa Inggris, proses *stemming* digunakan untuk menghilangkan sufiks (Agusta, 2009).

Algoritma Nazief-Adriani dikembangkan pertama kali oleh Bobby Nazief dan Mirna Adriani. Algoritma ini berdasarkan pada aturan morfologi Bahasa Indonesia yang luas, yang dikumpulkan menjadi satu group dan dienkapsulasi pada imbuhan (*affixes*) yang diperbolehkan (*allowed affixes*) dan imbuhan yang tidak diperbolehkan (*disallowed affixes*). Algoritma ini menggunakan kamus kata dasar dan mendukung *recording*, yakni penyusunan kembali kata-kata yang mengalami proses *stemming* berlebih (Susyanto, 2017).

Algoritma yang dibuat oleh Bobby Nazief dan Mirna Adriani memiliki tahapan sebagai berikut (Susyanto, 2017):

1. Cari kata yang akan di *stem* dalam kamus. Jika ditemukan maka diasumsikan bahwa kata tersebut adalah *root word*. Maka algoritma berhenti.
2. *Inflection Suffixes* (“-lah”, “-kah”, “-ku”, “-mu”, atau “-nya”) dibuang. Jika berupa particles (“-lah”, “-kah”, “-tah” atau “-pun”) maka langkah ini diulangi lagi untuk menghapus *Possesive Pronouns* (“-ku”, “-mu”, atau “-nya”), jika ada.
3. Hapus *Derivation Suffixes* (“-i”, “-an” atau “-kan”). Jika kata ditemukan di kamus, maka algoritma berhenti. Jika tidak maka ke langkah 3a.
  - 1) Jika “-an” telah dihapus dan huruf terakhir dari kata tersebut adalah “-k”, maka “-k” juga ikut dihapus. Jika kata tersebut ditemukan dalam kamus maka algoritma berhenti. Jika tidak ditemukan maka selanjutnya lakukan langkah 3b.
  - 2) Akhiran yang dihapus (“-i”, “-an” atau “-kan”) dikembalikan, lanjut ke langkah 4.
4. Hapus *Derivation Prefix*. Jika pada langkah 3 ada sufiks yang dihapus maka lanjutkan ke langkah 4a, jika tidak lanjutkan ke langkah 4b.
  - 1) Periksa tabel kombinasi awalan akhiran yang tidak diijinkan. Jika ditemukan maka algoritma berhenti, jika tidak lanjutkan ke langkah 4b.
  - 2) For  $i = 1$  to 3, tentukan tipe awalan kemudian hapus awalan. Jika *root word* belum juga ditemukan lakukan langkah 5, jika sudah maka algoritma berhenti. Catatan: jika awalan kedua sama dengan awalan pertama algoritma berhenti.
5. Melakukan *Recording*.
6. Jika semua langkah telah selesai tetapi tidak juga berhasil maka kata awal diasumsikan sebagai *root word*. Proses selesai.

### 3.3 Term Frequency (tf)

TF (*Term Frequency*) adalah frekuensi dari kemunculan sebuah *term* dalam dokumen yang bersangkutan. Semakin besar jumlah kemunculan suatu *term* (TF tinggi) dalam dokumen, semakin besar pula bobotnya atau akan memberikan nilai kesesuaian yang semakin besar (Rajaraman, 2011).

Variasi skema pembobotan *tf-idf* sering digunakan oleh mesin pencari sebagai alat utama dalam penilaian dan peringkat relevansi dokumen yang diberikan permintaan pengguna (Rajaraman, 2011).

TF merupakan salah satu skema pembobotan waktu paling populer saat ini; 83% dari sistem rekomendasi berbasis teks di perpustakaan digital menggunakan TF (Breitinger, 2015). Pada *Term Frequency* (TF), terdapat beberapa jenis formula yang dapat digunakan :

1. TF biner (binary TF), hanya memperhatikan apakah suatu kata atau *term* ada atau tidak dalam dokumen, jika ada diberi nilai satu (1), jika tidak diberi nilai nol (0).

2. TF murni (*raw TF*), nilai TF diberikan berdasarkan jumlah kemunculan suatu *term* di dokumen. Contohnya, jika muncul lima (5) kali maka kata tersebut akan bernilai lima (5).
3. TF logaritmik, hal ini untuk menghindari dominansi dokumen yang mengandung sedikit *term* dalam *query*, namun mempunyai frekuensi yang tinggi.
4. Perhitungan TF
 
$$TF = \begin{cases} 1 + \log_{10}(f_{t,d}), & f_{t,d} > 0 \\ 0, & f_{t,d} = 0 \end{cases} \quad (1)$$
 Dimana nilai  $f_{t,d}$  (1) adalah frekuensi term (t) pada dokumen (d). Jadi jika suatu kata atau *term* terdapat dalam suatu dokumen sebanyak 5 kali maka diperoleh bobot =  $1 + \log(5) = 1.699$ . Tetapi jika *term* tidak terdapat dalam dokumen tersebut, bobotnya adalah nol (0).
5. TF normalisasi (2), menggunakan perbandingan antara frekuensi sebuah *term* dengan nilai maksimum dari keseluruhan atau kumpulan *term* yang ada pada suatu dokumen.

$$TF = 0,5 + 0,5 \times \left[ \frac{f_{td}}{\max\{f_{t',d}, t', d \in d\}} \right] \quad (2)$$

Penelitian ini menggunakan formula TF biner yang menyatakan jumlah berapa banyak keberadaan suatu *term* dalam satu dokumen. Nilai TF biner dihitung dengan formula (Manning, 2009) berikut :

$$TF = \frac{\text{jumlah frekuensi kata terpilih}}{\text{jumlah kata}} \quad (3)$$

## 4. PERANCANGAN APLIKASI

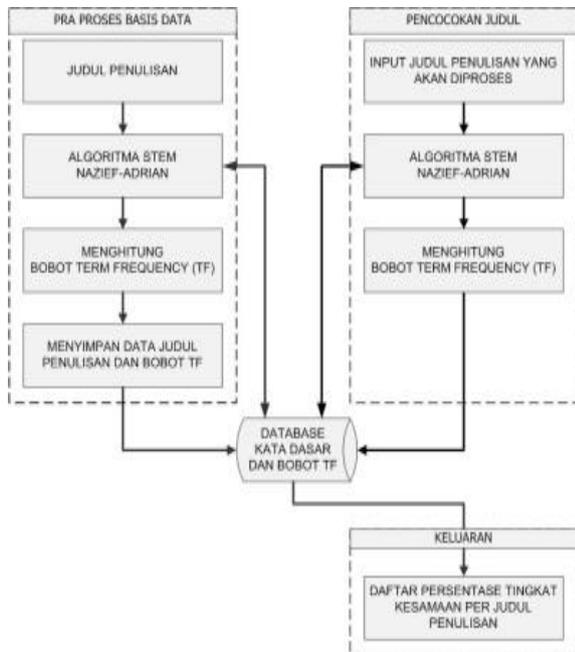
Dalam tahap ini akan dibahas mengenai perancangan sistem yang dibuat. Perancangan sistem merupakan rancangan yang dibuat sebelum perangkat lunak tersebut dibangun. Perancangan ini menggunakan bantuan alur sistem.

### 4.1 Pra-proses Basis Data

Pada proses ini yang pertama adalah memasukkan judul dari penulisan berbahasa Indonesia dan juga abstraksi atau kata kunci dari judul yang akan dimasukkan, lalu yang kedua itu ada proses algoritma *stemming* Nazief-Adriani yang dimana di algoritma ini akan menghilangkan kata imbuhan dari abstraksi atau kata kunci yang tadi telah dimasukkan seperti kata imbuhan -ber, -meng, -an, dan masih banyak lagi sesuai dengan algoritma tersebut. Pada tahap ketiga adalah menghitung bobot *term frequency* (tf) yang dimana setelah di *stemming* kata kuncinya menggunakan algoritma Nazief-Adrian maka akan dihitung *tf* kata kunci terhadap judul penulisan.

Pada tahap akhir yaitu menyimpan data seperti judul penulisan, kata dasar dan juga bobot *tf* yang tadi telah dihitung, lalu dimasukkan kedalam *database* yang berisikan data yang telah dimasukkan, kemudian algoritma *stemming* Nazief-Adrian juga dimasukkan

kedalam *database*, untuk mengambil kata dasar yang telah di *stemming* dan akan dicari kesamaan kata saat proses kesamaan pada *database* judul penulisan.



Gambar 1. Alur Pendeteksi Kesamaan Kata

Pada gambar 1 dapat terlihat perancangan alur sistem yang akan menjelaskan tentang alur dari sistem ini. Pada rancangan alur sistem ini terdapat dua proses, yang pertama itu adalah pra-proses basis data yang artinya itu adalah pada proses ini akan memproses data yang akan masuk ke *database*.

4.2 Perancangan Database

Untuk meng-inputkan data judul penulisan sebagai data pembanding maka diperlukan rancangan *database* yang dapat terlihat pada tabel 1.

Terdapat 5 *field* yang digunakan pada rancangan aplikasi ini, diantaranya *field* : *Id* untuk menyimpan kode identitas dari satu data judul penulisan, *judul* untuk menyimpan data judul penulisan, *total\_kata* untuk menyimpan banyaknya kata pada judul penulisan dan *kata\_dasar*, untuk menyimpan daftar kata dasar yang terdapat pada judul penulisan dengan *delimiter* berupa koma (,).

Tabel 1. Tabel Data

Field Name	Data Type
ID	AutoNumber
Judul	Text
Total_Kata	Number
Kata_Dasar	Memo

Untuk menyimpan master kata dasar menggunakan tabel 2. *Id* merupakan isian data identitas dari kata dasar, *kata\_dasar* merupakan isian kata dasar yang baku dalam

bahasa Indonesia dan *tipe* merupakan isian dari jenis kata : benda (nomina), sifat (adjektiva) dan kerja (verba).

Tabel 2. Kata Dasar

Field Name	Data Type
ID	AutoNumber
Kata_dasar	Text
Tipe	Text

Sebagai contoh kata dasar “informasi” bertipe nomina atau kata benda, “informatif” bertipe adjektiva atau kata sifat dan “ingat” bertipe verba atau kata kerja.

Tabel 3. Stop Word

Field Name	Data Type
ID	AutoNumber
Stop_Word	Text

Tabel 3 digunakan untuk menyimpan kata sebagai batas pada kalimat sehingga dapat kata yang akan di *stemming*. *Id* merupakan isian identitas dari kata pembatas dan *stop\_word* merupakan isian dari kata pembatas. Contoh kata pembatasnya : “yang”, “yakin”, “dan”, “dapat”, dan sebagainya

Tabel 4. Data Top

Field Name	Data Type
ID	AutoNumber
Id_Data	Number
Top_Kata	Text
Jumlah	Number
TF	Number

Tabel 4 digunakan untuk menyimpan potongan kata yang mengandung arti dari kalimat dan nilai *tf* berdasarkan judul penulisan yang diinputkan. *Id* merupakan isian identitas kata dasar, *id\_data* merupakan identitas judul penulisan, *top\_kata* merupakan isian kata dasar, *jumlah* merupakan isian jumlah kata dasar tersebut dalam judul penulisan dan *tf* merupakan isian *term frequency* dari kata dasar. Sebagai contoh judul penulisan “Aplikasi Perancangan Pengambilan Keputusan dengan Penentuan Korelasi Antar Produk Menggunakan Teknik Data Mining” jumlah total kata yang mengandung arti sebanyak 10, *Top\_Kata* :”aplikasi”, “perancangan”, “pengambilan”, “keputusan”, “penentuan”, “korelasi”, “produk”, “teknik”, “data” dan “mining”. Nilai *tf* didapatkan dari banyaknya kata telah dilakukan pemotongan dari judul penulisan dibagi dengan jumlah kata pada judul penulisan tersebut, misal kata “aplikasi” hanya ada 1 kata pada judul penulisan tersebut maka nilai *tf* dari kata “aplikasi” sebesar  $1/10 = 0.1$ .

5. IMPLEMENTASI

Aplikasi ini dibuat dengan tujuan untuk membantu pengguna yang ingin menentukan atau mencari kesamaan judul penulisan berbahasa Indonesia yang ada di dalam *database*. Aplikasi ini digunakan dengan

memasukkan judul penulisan berbahasa Indonesia. Judul penulisan tersebut lalu dihitung kesamaan kata dasar dan dihitung nilai persentasenya atau *term frequency*.



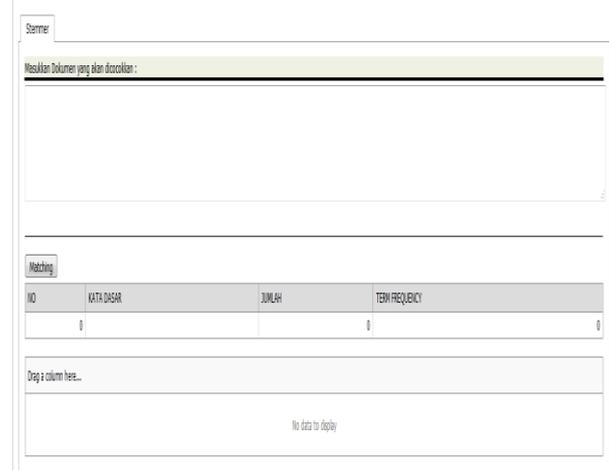
Gambar 2. Tampilan Input Judul Penulisan

Pada gambar 2 terlihat tampilan untuk inputan judul penulisan yang ingin dimasukkan secara langsung. Tampilan ini untuk menginputkan data yang dapat dilakukan dengan menggunakan dua cara menginputkan judul penulisan atau dengan menggunakan *import file* dari excel seperti pada gambar 3. Pada tampilan ini terdapat tombol simpan data yang akan menyimpan dokumen atau judul penulisan yang ingin ditambahkan ke *database*. Dan tombol *refresh* untuk memastikan bahwa data dokumen telah tersimpan dalam *database*.



Gambar 3. Tampilan Import file

Pada gambar 3 terlihat tampilan *import* dari *file* excel yang fungsinya sama seperti input judul penulisan. Pada tampilan ini terdapat *button* telusuri yang fungsinya untuk memilih *file* excel yang ingin dimasukkan dan tombol lihat excel yaitu untuk melihat isi dari *file* excel yang telah dipilih dan ada tombol simpan data yang digunakan untuk menyimpan data excel yang telah dipilih, lalu ada tombol *refresh* yang sama seperti pada tampilan input dokumen.



Gambar 4. Tampilan Pencarian Kesamaan Judul Penulisan

Gambar 4 merupakan tampilan untuk proses pencarian kesamaan judul penulisan yang ingin dicari kesamaan kata pada *database* judul penulisan. Dalam tampilan tersebut terdapat sebuah kolom untuk menginputkan judul penulisan yang akan dicarikan kesamaan kata, tombol *matching* berfungsi untuk melakukan proses pencarian kesamaan kata pada *database* judul penelitian yang telah diinputkan. Tabel yang terdapat dalam tampilan berguna untuk menampilkan kata dasar yang terdapat pada inputan judul penulisan dan *term frequency*, seperti pada gambar 5.

NO	KATA DASAR	JUMLAH	TERM FREQUENCY
1	DATA	1	0,5
2	MINDING	1	0,5

ID	JUDUL	DATA	MINDING	PERSENTASE
1	PENYINGKIHAN SISTEM INFORMASI PENJAJAN KINERJA DOSEN DAN FORUM MAHASISWA BERBASIS WEBSITE			0
2	APLIKASI AUGMENTED REALITY PENGENALAN ALAT KESELAMATAN DI ATAS KAPAL BERBASIS ANDROID			0
3	PENYINGKIHAN SISTEM INFORMASI PENJAJAN PRODUK SUGIH BERBASIS WEBSITE			0
4	APLIKASI AUGMENTED REALITY PENGENALAN ALAT MUSIK TRADISIONAL SUMATERA UTARA			0
5	PENYINGKIHAN APLIKASI UNTUK MELIHAT DESAIN PERABOTAN RUMAH TANGGA MENGGUNAKAN AUGMENTED REALITY BERBASIS ANDROID			0
6	PENYINGKIHAN APLIKASI PENGENALAN MEWANI MARALIA DENGAN MENGGUNAKAN METODE AUGMENTED REALITY BERBASIS ANDROID			0
7	APLIKASI LARJALAU DAERAH PULAU JAJAH MENGGUNAKAN SMARTPHONE ANDROID			0
8	PENYINGKIHAN SISTEM MEDIA PROMOSI DESAIN INTERIOR BERBASIS WEBSITE			0
9	IMPLEMENTASI DATA MINDING PADA FEEDBACK BARY SHOP DENGAN MENGGUNAKAN ALGORITMA ANDROID	1	1	100
10	PERBANDINGAN SOFTWARE APLIKASI ADOBE PHOTOSHOP CS4 DENGAN SOFTWARE APLIKASI ADOBE PHOTOSHOP ELEMENTS 6.0			0
11	PERBANDINGAN MAIL SERVER MENGGUNAKAN CTRADSL DAN MAIL CLIENT MENGGUNAKAN MOZILLA THUNDERBIRD PADA LINUX UBUNTU			0
12	RANCANGAN BAGIAN HOTSPOT DENGAN MIKROTIK ROUTER OS GUNA MENBERIKAN LAJAHAN BERBASIS IT BAGI PENGUNJUNG KOLAM REWANG ATLANTIS ANCOL			0
13	PENGISI AIR OTOMATIS MENGGUNAKAN LCD KARAKTER BERBASIS MIKROKONTROLER AT89C51			0
14	SIMULASI PERBANDINGAN JARINGAN KOMPUTER CLIENT SERVER MENGGUNAKAN TOPOLOGI STAR DAN CIRCULAR PADA PACKET TRACER			0
15	RANCANGAN BANGUNAN JARINGAN WIRELESS LAN PADA DEPOT PERUSAHA TANGGUNG MANDIRI GUNA DAPAT BERKOMUNIKASI ANTARA DEPOT DI LINGKUNGAN DALAM BESAR KARANTINA PELENGKAPAN TANGKAP			0
16	PILU SEBAGAI PERSEKUTUHAN OPERASIONAL MEKANISME COMEYOR PENJAJAN BERKONDISI BERMOTOR DI PT SHF INDONESIA			0

Gambar 5. Tampilan nilai *tf* dan hasil pencarian kesamaan

Beberapa pengujian telah dilakukan untuk mengetahui sistem berjalan sesuai dengan perancangan. Pengujian ini dilakukan untuk mengetahui tingkat persentase kesamaan judul penulisan yang diinputkan dengan *database* dokumen atau judul penulisan. Pengujian dilakukan terhadap 5 judul penulisan sebagai berikut :

1. Judul Penulisan : Aplikasi Perancangan Pengambilan Keputusan dengan Penentuan Korelasi Antar Produk Menggunakan Teknik Data Mining

Stemmer

Mencarikan judul yang akan diocokkan :  
 APLIKASI PERANCANGAN PENGAMBILAN KEPUTUSAN DENGAN PENENTUAN KORELASI ANTAR PRODUK MENGGUNAKAN TEKNIK DATA MINING

2.07 sec.

NO	KATA DASAR	JUMLAH	TF FREQUENCY
1	APLIKASI	1	0.1
2	DATA	1	0.1
3	KEPUTUSAN	1	0.1
4	KORELASI	1	0.1
5	MINGING	1	0.1
6	PENENTUAN	1	0.1
7	PENGAMBILAN	1	0.1
8	PERANCANGAN	1	0.1
9	PRODUK	1	0.1
10	TEKNIK	1	0.1

Gambar 6. Hasil Kata Dasar dari Pengujian 1

Gambar 6 menampilkan hasil kata dasar dari judul penulisan pada pengujian ke-1, “Aplikasi Perancangan Pengambilan Keputusan dengan Penentuan Korelasi Antar Produk Menggunakan Teknik Data Mining”. Dari judul tersebut setelah dilakukan proses *stemming* menggunakan algoritma Nazief-Adriani, maka dihasilkan kata dasar dan nilai *tf* nya berikut ini : “aplikasi”, “data”, “keputusan”, “korelasi”, “mining”, “keputusan”, “penentuan”, “pengambilan”, “perancangan”, “produk” dan “teknik”.

ID	JUDUL	APLIKASI	DATA	KEPUTUSAN	KORELASI	MINGING	PENENTUAN	PENGAMBILAN	PERANCANGAN	PRODUK	TEKNIK	PERSENTASE
38	DATA MINING PENENTUAN JURUSAN CALON MAHASISWA LULUSAN SMK MENGGUNAKAN ALGORITMA K-NEAREST NEIGHBOR CLASSIFIER PADA UNIVERSITAS		1			1	1					30
40	PERANCANGAN APLIKASI UNTUK PENGOLOHAN DATA PASIEN PADA KLINIK DR. HURUS BERBASIS DESKTOP MENGGUNAKAN VB.NET	1	1						1			30
61	APLIKASI ANDROID TEKNIK PENYERAPAN BELAJAR MENGGUNAKAN KODE FLASH										1	20
118	APLIKASI PENGOLOHAN DATA DAN LAPORAN PENGOLOHAN OBAT TI KLINIK PRATIAMA AL FALAH MENGGUNAKAN VISUAL STUDIO 2010	1	1									20
123	APLIKASI PENGOLOHAN DATA TANPA WAKAF KANAK RUMAH BERBENTUK BENTANG JAKARTA DENGAN MENGGUNAKAN SOFTWARE VISUAL BASIC.NET 2010 DAN SQL SERVER 2014		1	1								20
140	APLIKASI PANDUAN TEKNIK PEREDA MOTOR BERBASIS ANDROID DENGAN MENGGUNAKAN ANDROID STUDIO										1	20

Gambar 7. Hasil Kesamaan judul penulisan dari Pengujian 1

Hasil kesamaan judul penulisan seperti gambar 7 didapatkan dengan mencari kesamaan kata dasar dengan data judul penulisan yang ada dalam *database*. Persentase terbesar 30% didapatkan pada 2 judul penulisan yaitu “Data Mining Penentuan Jurusan Mahasiswa Lulusan SMK Menggunakan Algoritma K-Nearest Neighbor Classifier Pada Universitas” dan “Perancangan Aplikasi Untuk Pengolahan Data Pasien Pada Klinik DR. Hurus Berbasis Desktop Menggunakan VB.NET” dengan kesamaan pada 5 kata yaitu kata “aplikasi”, “data”, “mining”, “penentuan” dan “perancangan”. Proses kesamaan kata dengan membandingkan 1561 judul penulisan pada *database* membutuhkan waktu selama 2.07 detik.

- Judul Penulisan : Aplikasi Lokasi Wisata Bali Menggunakan Teknologi Augmented Reality Berbasis Android dan PHP

Stemmer

Mencarikan judul yang akan diocokkan :  
 APLIKASI LOKASI WISATA BALI MENGGUNAKAN TEKNOLOGI AUGMENTED REALITY BERBASIS ANDROID DAN PHP

3.01 sec.

NO	KATA DASAR	JUMLAH	TF FREQUENCY
1	ANDROID	1	0.1
2	APLIKASI	1	0.1
3	AUGMENTED	1	0.1
4	BALI	1	0.1
5	BERBASIS	1	0.1
6	LOKASI	1	0.1
7	PHP	1	0.1
8	REALITY	1	0.1
9	TEKNOLOGI	1	0.1
10	WISATA	1	0.1

Gambar 8. Hasil Kata Dasar dari Pengujian 2

Gambar 8 menampilkan kata dasar pada judul penulisan pengujian ke-2, “Aplikasi Lokasi Wisata Bali Menggunakan Teknologi Augmented Reality Berbasis Android Dan PHP” dengan hasil *stemming* didapatkan kata dasar beserta dengan nilai *tf* nya : “android”, “aplikasi”, “augmented”, “bali”, “berbasis”, “lokasi”, “php”, “reality”, “teknologi” dan “wisata”.

ID	JUDUL	ANDROID	APLIKASI	AUGMENTED	BALI	BERBASIS	LOKASI	PHP	REALITY	TEKNOLOGI	WISATA	PERSENTASE
137	APLIKASI BROSUR PEMASARAN PERUMAHAN BOGOR VIEW MENGGUNAKAN TEKNOLOGI AUGMENTED REALITY BERBASIS ANDROID	1	1	1	1	1	1	1	1	1		60
379	APLIKASI MEDIA PEMBELAJARAN MATAHARI, BULAN, GERHANA MENGGUNAKAN TEKNOLOGI AUGMENTED REALITY BERBASIS ANDROID	1	1	1	1	1	1	1	1	1		60
1094	APLIKASI ANIME ONE PIECE MENGGUNAKAN TEKNOLOGI AUGMENTED REALITY BERBASIS MOBILE ANDROID	1	1	1	1	1	1	1	1	1		60
1168	APLIKASI PENGENALAN TATA SURYA BERBASIS ANDROID MENGGUNAKAN TEKNOLOGI AUGMENTED REALITY (AR)	1	1	1	1	1	1	1	1	1		60
1502	PEMBUATAN APLIKASI PENGENALAN BINTANG DAN PLANET BERBASIS ANDROID MENGGUNAKAN TEKNOLOGI AUGMENTED REALITY	1	1	1	1	1	1	1	1	1		60
1537	PEMBUATAN APLIKASI PENGENALAN TAHAP-TAHAP IBADAH HAJI BERBASIS ANDROID MENGGUNAKAN TEKNOLOGI AUGMENTED REALITY	1	1	1	1	1	1	1	1	1		60
16	APLIKASI PENGOLOHAN TEKNIK PEREDA MOTOR BERBASIS ANDROID DENGAN MENGGUNAKAN ANDROID STUDIO	1	1	1	1	1	1	1	1	1		50

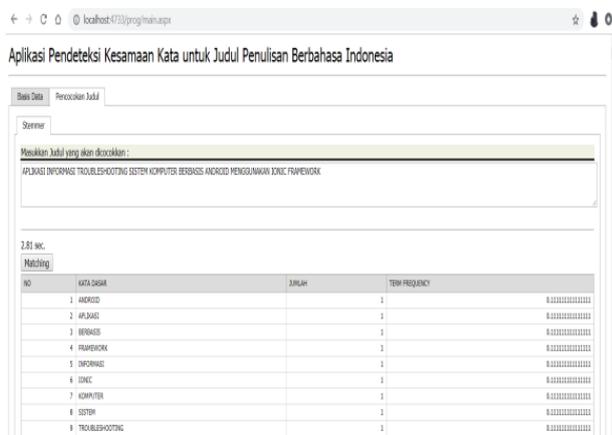
Gambar 9. Hasil Kesamaan judul penulisan dari Pengujian 2

Pada gambar 9 ditampilkan hasil kesamaan kata dasar yang dibandingkan dengan *database* judul penulisan. Persentase terbesar 60% didapatkan dari 6 judul penulisan di antaranya : “Aplikasi Brosur Pemasaran Perumahan Bogor View Menggunakan Teknologi Augmented Reality Berbasis Android”, “Aplikasi Media Pembelajaran Matahari, Bulan, Gerhana Menggunakan Teknologi Augmented Reality Berbasis Android”, “Aplikasi Anime One Piece Menggunakan Teknologi Augmented Reality Berbasis Mobile Android”, “Aplikasi Pengenalan Tata Surya Berbasis Android Menggunakan Teknologi Augmented Reality (AR)”, “Pembuatan Aplikasi Pengenalan Bintang Dan Planet Berbasis Android Menggunakan Teknologi Augmented Reality” dan “Pembuatan Aplikasi Pengenalan Tahap-Tahap Ibadah Haji Berbasis Android Menggunakan Teknologi Augmented Reality”. Kesamaan kata didapatkan pada 5 kata dasar, yaitu : “android”, “aplikasi”, “augmented”, “berbasis”, “reality” dan “teknologi”. Proses kesamaan kata dengan membandingkan dengan 1561 judul

penulisan pada *database* membutuhkan waktu selama 3.01 detik.

3. Judul Penulisan : Aplikasi Informasi Troubleshooting Sistem Komputer Berbasis Android Menggunakan Ionic Framework

Gambar 10 menampilkan kata dasar pada judul penulisan pengujian ke-3, “Aplikasi Informasi Troubleshooting Sistem Komputer Berbasis Android Menggunakan Ionic Framework” dengan hasil *stemming* didapatkan kata dasar beserta dengan nilai *tf*-nya, yaitu : “android”, “aplikasi”, “berbasis”, “framework”, “informasi”, “ionic”, “komputer”, “sistem” dan “troubleshooting”.



Gambar 10. Hasil Kata Dasar dari Pengujian 3

ID	JUDUL	ANDROID	APLIKASI	BERBASIS	FRAMEWORK	INFORMASI	IONIC	KOMPUTER	SISTEM	TROUBLESHOOTING	PERSENTASE
444	APLIKASI EDUKASI DAN INFORMASI LABORATORIUM LANJUT SISTEM KOMPUTER UNIVERSITAS GUNADARMA BERBASIS ANDROID	1	1	1	1	1	1	1	1	1	66.6666666666667
648	APLIKASI INFORMASI PULAU SULAWESI BERBASIS ANDROID MENGGUNAKAN IONIC FRAMEWORK	1	1	1	1	1	1	1	1	1	66.6666666666667
46	PERUBAHAN APLIKASI SISTEM INFORMASI PERSEKAWAN BERBASIS ANDROID	1	1	1	1	1	1	1	1	1	55.5555555555556
331	APLIKASI SISTEM INFORMASI PENETAPAN PUSAT PERBELAJARAN MAL DI TANGERANG BERBASIS ANDROID	1	1	1	1	1	1	1	1	1	55.5555555555556
561	PERUBAHAN APLIKASI SISTEM INFORMASI PROSESING BERBASIS WEBSTE MENGGUNAKAN FRAMEWORK LARAVEL	1	1	1	1	1	1	1	1	1	55.5555555555556
562	APLIKASI SISTEM INFORMASI MANAJEMEN SAMPUL SITE BERBASIS WEBSTE MENGGUNAKAN BOOTSTRAP DAN FRAMEWORK LARAVEL	1	1	1	1	1	1	1	1	1	55.5555555555556
153	PERUBAHAN APLIKASI INFORMASI PENGELOMAN SISTEM DATA SURVA DAN PERHENTIAN GERAKAN BERBASIS ANDROID	1	1	1	1	1	1	1	1	1	55.5555555555556

Gambar 11. Hasil Kesamaan judul penulisan dari Pengujian 3

Pada gambar 11 ditampilkan hasil kesamaan kata dasar yang dibandingkan dengan *database* judul penulisan. Persentase terbesar 66,6% didapatkan dari 2 judul penulisan di antaranya : “Aplikasi Edukasi Dan Informasi Laboratorium Lanjut Sistem Komputer Universitas Gunadarma” dan “Aplikasi Informasi Pulau Sulawesi Berbasis Android Menggunakan Ionic Framework” Kesamaan kata didapatkan pada 8 kata dasar, yaitu : “android”, “aplikasi”, “berbasis”, “framework”, “informasi”, “ionic”, “komputer” dan “sistem”. Proses kesamaan kata menbandingkan dengan 1561 judul penulisan pada *database* membutuhkan waktu selama 2.81 detik.

4. Judul Penulisan : Pengelolaan Dana Kas Kecil Pada PT. PLN (PERSERO) UJT Pulogadung Jakarta Timur



Gambar 12. Hasil Kata Dasar dari Pengujian 4

Gambar 12 menampilkan kata dasar pada judul penulisan pengujian ke-4, “Pengelolaan Dana Kas Kecil Pada PT. PLN (PERSERO) UJT Pulo Gadung Jakarta Timur” dengan hasil *stemming* didapatkan kata dasar beserta dengan nilai *tf* nya : “dana”, “jakarta”, “kas”, “pengelolaan”, “persero”, “ptpln”, “pulogadung”, “timur” dan “ujt”.

ID	JUDUL	DANA	JAKARTA	KAS	PENGELOMAN	PERSERO	PT	PLN	PULOGADUNG	TIMUR	UJT	PERSENTASE
112	APLIKASI INFORMASI KULINER DI PULO GEBANG JAKARTA TIMUR BERBASIS ANDROID	1	1	1	1	1	1	1	1	1	1	20
207	SISTEM PENGGAJIAN PADA PT. TASPEN (PERSERO)	1	1	1	1	1	1	1	1	1	1	20
557	APLIKASI LOCATION BASED SERVICE (LBS) UNTUK MENCARI LOKASI SPBU DI JAKARTA TIMUR BERBASIS PLATFORM ANDROID	1	1	1	1	1	1	1	1	1	1	20
564	APLIKASI INFORMASI PENCARIAN LOKASI KANTOR POLISI DI WILAYAH JAKARTA TIMUR BERBASIS ANDROID	1	1	1	1	1	1	1	1	1	1	20
644	APLIKASI PENCARIAN SPBU DI JAKARTA TIMUR DENGAN MENGGUNAKAN LBS (LOCATION BASED SERVICES)	1	1	1	1	1	1	1	1	1	1	20
805	APLIKASI PENCARIAN LOKASI DAN INFORMASI MALL DI DAERAH JAKARTA TIMUR BERBASIS ANDROID	1	1	1	1	1	1	1	1	1	1	20
940	PENCARIAN LOKASI BIMBINGAN BELAJAR MENGGUNAKAN METODE LOCATION-BASED SERVICES PADA APLIKASI ANDROID DI WILAYAH JAKARTA TIMUR	1	1	1	1	1	1	1	1	1	1	20
108	APLIKASI PENCARIAN RUMAH PUBLIK TERPADU RAMAH ANAK (RPTRA) DAERAH KOTAMADYA JAKARTA TIMUR DENGAN METODE LOCATION BASED SERVICE BERBASIS ANDROID	1	1	1	1	1	1	1	1	1	1	20

Gambar 13. Hasil Kesamaan judul penelitian dari Pengujian 4

Pada gambar 13 ditampilkan hasil kesamaan kata dasar yang dibandingkan dengan *database* judul penulisan. Persentase terbesar 20% didapatkan dari 8 judul penulisan yaitu : “Aplikasi Informasi Kuliner Di Pulo Gebang Jakarta Timur Berbasis Android”, “Sistem Penggajian Pada PT. TASPEN (Persero)”, “Aplikasi Location Based Service (LBS) Untuk Mencari Lokasi SPBU Di Jakarta Timur Berbasis Platform Android”, “Aplikasi Informasi Pencarian Lokasi Kantor Polisi Di Wilayah Jakarta Timur Berbasis Android”, “Aplikasi Pencarian SPBU Di Jakarta Timur Dengan Menggunakan LBS (Location Based Services)”, “Aplikasi Pencarian Lokasi Dan Informasi Mall Di Daerah Jakarta Timur Berbasis Android”, “Pencarian Lokasi Bimbingan Belajar Menggunakan Metode Location-Based Services Pada Aplikasi Android Di Wilayah Jakarta Timur” dan “Aplikasi Pencarian Ruang Publik Terpadu Ramah Anak (RPTRA) Daerah Kotamadya Jakarta Timur Dengan Metode Location Based Services Berbasis Android” . Kesamaan kata didapatkan pada 4 kata dasar, yaitu : “jakarta”, “persero”, “pt” dan “timur”. Proses kesamaan kata membutuhkan waktu selama 1.55 detik

## 5. Judul Penulisan : Analisis Penilaian Kinerja Dengan Rasio Keuangan Pada PT. Surveyor Indonesia

Gambar 14 menampilkan kata dasar pada judul penulisan pengujian ke-5, “Analisis Penilaian Kinerja Dengan Rasio Keuangan Pada PT. Surveyor Indonesia” dengan hasil *stemming* didapatkan kata dasar beserta dengan nilai *tf* nya : “analisis”, “indonesia”, “keuangan”, “kinerja”, “penilaian”, “pt”, “rasio” dan “surveyor”.

**Gambar 14. Hasil Kata Dasar dari Pengujian 5**

ID	JUDUL	ANALISIS	INDONESIA	KEUANGAN	KINERJA	PENILAIAN	PT	RASIO	SURVEYOR	PERSENTASE
41	PENGEMBANGAN SISTEM INFORMASI PELAJAN KINERJA DOSEN DAN FORUM MAHASISWA BERBASIS WEB SITE				1	1				25
54	APLIKASI E-LEARNING MANAJEMEN PIUTANG DAN RASIO KEUANGAN BERBASIS WEB PADA LABORATORIUM MANAJEMEN MENENGAH				1			1		25
386	ANALISIS KINERJA SOFTWARE AUDIO EDITING SOFTWARE : SOUND FORGE & GOLD WAVE	1			1					25
603	OPTIMASI PROXY SERVER DI PT. SKF INDONESIA			1				1		25
679	RANCANGAN DAN IMPLEMENTASI FIREWALL DALAM PENGEMBANGAN INFRASTRUKTUR BERJARINGAN KOMPUTER DI PT. COMPUTRADE TECHNOLOGY INDONESIA			1				1		25
1128	PENGEMBANGAN CENTRALIZE MOBILE DATA PROTECTION UNTUK DATA YANG BERSIFAT PENTING DAN RAHASIA PADA PT. COMPUTRADE TECHNOLOGY INDONESIA			1				1		25
9	APLIKASI SAKTI-SAKTI DI INDONESIA BERBASIS ANDROID 4.1.2			1						12.5
18	APLIKASI DAFTAR HAJAT PAKET INTERNET DARI OPERATOR SELULER DI INDONESIA			1						12.5

**Gambar 15. Hasil Kesamaan judul penelitian dari Pengujian 5**

Pada gambar 15 ditampilkan hasil kesamaan kata dasar yang dibandingkan dengan *database* judul penulisan. Persentase terbesar 25% didapatkan dari 6 judul penulisan yaitu : “Pengembangan Sistem Informasi Penilaian Kinerja Dosen Dan Forum Mahasiswa Berbasis Web Site”, “Aplikasi E-Learning Manajemen Piutang Dan Rasio Keuangan Berbasis Web Pada Laboratorium Manajemen Menengah”, Analisis Kinerja Software Audio Editing Software : Sound Forge & Gold Wave”, “Optimasi Proxy Server di PT. SKF Indonesia”, “Rancangan Dan Implementasi Firewall Dalam Pengembangan Infrastruktur Jaringan Komputer Di PT. Computrade Technology Indonesia” dan “Perancangan Centralize Mobile Data Protection Untuk Data Yang Bersifat Penting Dan Rahasia Pada PT. Computrade Technology Indonesia”. Kesamaan kata didapatkan pada 6 kata dasar, yaitu : “analisis”, “indonesia”, “keuangan”, “kinerja”, “penilaian” dan “rasio”. Proses kesamaan kata membutuhkan waktu selama 1.27 detik

## 6. KESIMPULAN

Telah dilakukan pengujian pada aplikasi pendeteksi kesamaan kata untuk judul penulisan berbahasa Indonesia ini menggunakan algoritma *stemming* Nazief-Adriani dan Term *Frequency* dengan *database* berjumlah 1561 judul penulisan berbahasa Indonesia.

Dari hasil pengujian dapat disimpulkan penelitian ini dapat membantu mahasiswa maupun dosen untuk mencari referensi judul penulisan berbahasa Indonesia yang ingin dibuat oleh mahasiswa agar tidak terjadi kesamaan atau *plagiarism* antara judul penulisan yang ingin dibuat.

Waktu komputasi untuk mencari kesamaan kata ditampilkan pada aplikasi ini, dengan waktu minimum kisaran 1.27 detik dan maksimum di kisaran 3.21 detik.

## 7. SARAN

Disarankan agar aplikasi ini dikembangkan dengan menampilkan hasil kesamaan dalam bentuk yang lebih interaktif seperti berbentuk grafik tingkat kesamaan. Untuk mempercepat proses komputasi disarankan dengan menggunakan proses secara paralel.

## 8. DAFTAR PUSTAKA

- Adriani, M., Jelita, A., Williams, HE., Tahaghoghi, SMM., Nazief, B. 2007. Stemming Indonesian : A Confix-stripping approach . *Journal ACM Transaction on Asian Language Information Processing (TALIP)*, Volume 6 Issue 4, USA.
- Agusta dan Ledy. 2009. Perbandingan Algoritma Stemming Porter dengan Algoritma Nazief & Adriani untuk Stemming Dokumen Teks Bahasa Indonesia. *Prosiding Konferensi Nasional Sistem dan Informatika*, November 14th, Bali, Indonesia.
- Azhar, F., Ernawati & Vatesia, A. 2014. Aplikasi Pendeteksi Kemiripan Pada Dokumen Teks Menggunakan Algoritma Nazief & Adriani dan Metode Cosine Similarity. *Jurnal Teknologi Informasi*, Volume 10 Nomor 1, ISSN : 1414-9999.
- Breitinger, C., Gipp, B., Langer, S. 2015. "Research-paper recommender systems: a literature survey" (Submitted manuscript). *International Journal on Digital Libraries*. 17 (4): 305–338. doi:10.1007/s00799-015-0156-0. ISSN 1432-5012.
- Kao dan Poteet, S. 2005. Text mining and natural language processing: introduction for the special issue. *SIGKDD Explor. Newsl*, vol. 7, no. 1, pp. 1-2.
- Khan, M.A., Aleem, A., Wahab, A dan Khan, MN. 2011. Copy detection in Urdu language documents using n-grams model. *Proceedings of the International Conference on Computer Networks and Information Technology*, July 11-13, Abbottabad, Pakistan, pp: 263-266
- Manning, Christopher D., Raghavan, P dan Schütze, H. 2009. *An Introduction to Information Retrieval*. Cambridge: Cambridge University
- Ohana, B., Tierney, B. 2011. Supervised Learning Methods for Sentiment Classification with RapidMiner. *RapidMiner Community Meeting And Conference*, RCOMM, pp. 1-8.

- Poernomo, B., Gunawan, Ir. 2015. Sistem Information Retrieval Pencarian Kesamaan Ayat Terjemahan Al Quran Berbahasa Indonesia Dengan Query Expansion Dari Tafsirnya, *Seminar Nasional "Inovasi dalam Desain dan Teknologi*, IdeaTech, ISSN : 2089-1121
- Rajaraman, A.; Ullman, J.D. 2011. "Data Mining". *Mining of Massive Datasets*. pp. 1–17. doi:10.1017/CBO9781139058452.002. ISBN 978-1-139-05845-2
- Simarangkir, MSH. 2017. Studi Banding Algoritma-Algoritma *Stemming* Untuk Dokumen Teks Bahasa Indonesia. *Jurnal Infokar*, Volume 1, ISSN : 2615-3645.
- Susyanto, T., Wahyudi, D., Nugroho, D. 2017. Implementasi Dan Analisis Algoritma *Stemming* Nazief & Adriani Dan Porter Pada Dokumen Berbahasa Indonesi. *Jurnal Ilmiah SINUS*, ISSN : 1693-1173