PENGUJIAN TINGKAT KEMIRIPAN SKRIPSI MAHASISWA DENGAN ALGORITMA GENETIKA MENGGUNAKAN POSI FORMULATION

Darwis Robinson Manalu

Program Studi Sistem Informasi, Universitas Methodist Indonesia
Jln Hang Tuah No 8 Medan
manaludarwis@gmail.com

Abstract

Measures the percentage similarity becomes important documents today because of the many digital documents in particular scientific work. Measurements were performed by using keywords solution of some of the documents that the user selected after the competition keywords. The process of determining keywords algorithm solution with genetics to produce the latest generation of the best with Jaccard Function. Then the similarity calculation process will be done using the keyword query comparing the solution with an existing document in a database that is in the post title, abstract, keywords and references. Similarity calculation method used is the Percentage of Similarity (POSI) Formulation. The number of keywords found on each document will be distributed in the whole of keywords that are found to produce a percentage of similarity. From the tests results obtained similarity of documents idDoc-661 amounted to 32.26%, 24.19% in the second sequence idDoc-665 and the third number of 19.35% on idDoc-663 and fourth at 12.90% in idDoc -667 and the latter by 11.29% in idDoc-666. Based on these tests can find out similarities with the other documents.

Keywords: *Document similarity, Genetic Algorithm (GA), Jaccard Function, POSI Formulation*

1. PENDAHULUAN

DEngan meningkatnya jumlah hasil penelitian maupun karya ilmiah lainnya dalam bentuk karya ilmiah digital terutama pada bidang akademik seperti skripsi, tesis, jurnal, prosiding dan sejenisnya, sehingga kemungkinan karya ilmiah dapat terjadi kemiripan seperti pada judul tulisan, abstrak, permasalahan, metode yang digunakan, objek penelitian, pembahasan dan hasilnya. Agar penulisan sebuah karya ilmiah tidak terjadi pengulangan maka perlu dilakukan antisipasi sejak dini. Pengujian kemiripan karya ilmiah merupakan pendeteksian kesamaan beberapa dokumen dengan membandingkan isi dokumen sehingga menghasilkan bobot atau nilai kemiripan dari karya ilmiah yang dibandingkan. Salah satu kegunaan perbandingan isi dokumen adalah untuk membantu pengguna dalam pengelompokan karya ilmiah dan juga memungkinkan pengguna mengetahui apakah isi karya ilmiah yang satu merupakan karya ilmiah yang pada dasarnya sama dengan karya ilmiah yang lain. Hal ini berfungsi untuk mengetahui apakah sebuah karya ilmiah mirip dengan karya ilmiah lain (Sihombing, 2010).

Pengujian kemiripan karya ilmiah ini dapat dilakukan dengan beberapa teknik, misalnya teknik pencarian informasi, teknik penghitungan statistik, atau dengan menggunakan informasi sintaktik dari kalimat perkalimatnya (Taufiq, 2013). Pendekatan-pendekatan tersebut tidaklah sempurna, masih terdapat beberapa kelemahan, misalnya penghitungan statistik yang membandingkan frekuensi kata dari dokumen satu dengan dokumen yang lain, tidak memperhatikan struktur kalimat. Sedangkan dalam teknik sintaktik kalimat, urutan kata dalam kalimat diperiksa unsur semantiknya dengan cara mengolah letak kata sesuai tatabahasanya atau dengan penggantian sebuah kata dengan sinonim dari kata tersebut. Teknik ini mempunyai kelemahan, yaitu setiap kata dikelompokkan pada label masing-masing untuk mengetahui struktur kalimat. Penelitian ini bertujuan mengembangkan pengujian kemiripan satu dokumen dengan dokumen lain yang berada dalam satu database. Proses yang akan dilakukan adalah dengan mengadakan kompetisi kata kunci untuk mendapatkan kata kunci solusi (*keyword solution*) yang ada pada sebuah dokumen menggunakan algoritma genetika metode *Jaccard*. Dalam pengujian sebuah karya ilmiah dapat dibandingkan dengan multi dokumen yang telah dikelompokkan dalam sebuah server atau pusat database. Dimana informasi terhadap sebuah dokumen tersebut sudah dimasukkan terlebih dahulu dalam bentuk digital seperti kata kunci, id dokumen, abstrak, judul, dan informasi penting yang dapat mewakili informasi dokumen tersebut seperti daftar pustaka atau referensi utama yang digunakan. Sehingga diharapkan dokumen yang diuji kemiripannya dapat menghasilkan persentase kemiripan antara dokumen yang dipilih oleh user dibandingkan dengan sekumpulan dokumen lainnya dalam database.

Maka pengguna karya ilmiah dapat terbantu dalam mengetahui isi dari sebuah dokumen/karya ilmiah tanpa harus membaca isi keseluruhan dokumen tersebut. Adapun rumusan masalah dalam penelitian ini adalah berapa persentase kemiripan sebuah dokumen jika dibandingkan dengan dokumen lain di dalam sebuah database dengan proses algoritma genetika menggunakan Perentage of Similarity (POSI) Formulation" adapun kajian yang dibahas adalah metode yang digunakan *Jaccard Function, k*arya ilmiah yang diuji adalah berupa dokumen jurnal yang telah memiliki format penulisan yang sama, Formula perhitungan pengukuran kemiripan dengan *POSI Formulation.* Data yang digunakan yang bersumber dari karya Ilmiah/Prosiding Seminar Nasional Ilmu Komputer (SNIKOM) APTIKOM Wilayah I Tahun 2013 serta pengujian kata kunci solusi dilakukan terhadap judul tulisan, kata kunci, abstrak dan referensi. Aplikasi yang dirancang digunakan secara multiuser dan menyediakan fasilitas pencarian dokumen pada aplikasi untuk memudahkan mengetahui isi karya ilmiah. Adapun tujuan penelitian ini adalah untuk mengetahui hasil persentase kemiripan sebuah karya ilmiah dengan karya ilmiah lainnya. Sedangkan manfaatnya adalah pengguna dapat mengetahui dengan cepat kemiripan karya ilmiah tanpa harus membaca keseluruhan isi dokumen. Adapun kontribusi penelitian yang dilakukan adalah:

1. Menambah salah satu cara untuk mengukur kemiripan dokumen berbasis teks dalam sebuah pusat data yang terdiri dari dokumen jurnal dan karya ilmiah lainnya.
2. Membuat clustering dalam database server untuk mempercepat proses pengukuran kemiripan menggunakan fungsi SQL.

3. Aplikasi yang dirancang berbasis GUI yang dapat dipergunakan secara multiuser dan menyediakan fasilitas search pada database dokumen.
4. Menggunakan referensi untuk menambah kata kunci dalam melakukan kompetisi kata kunci.

2. ALGORITMA GENETIKA

Algoritma genetika adalah salah satu cabang AI (*Artificial Intelligence)* yang merupakan model perhitungan yang diinspirasikan oleh teori evolusi. Algoritma tersebut mengkodekan solusi-solusi potensial untuk permasalahan yang ada pada struktur data berupa kromosom. Algoritma genetika umumnya dipandang sebagai fungsi optimisasi, meskipun jangkauan permasalahan yang telah diaplikasikan oleh genetika algoritma sangat luas, yaitu :

1. Al Biles menggunakan algoritma genetika untuk memfilter bagian yang baik dan buruk untuk improvisasi jazz.
2. Militer menggunakan algoritma genetika untuk mengembangkan persamaan untuk mendapatkan perbedaan di antara perputaran radar.
3. Perusahaan-perusahaan menggunakan algoritma genetika untuk memprediksikan pasar bursa.

Kebanyakan sistem AI simbolis adalah sangat statis. Biasanya digunakan hanya untuk memecahkan satu masalah khusus, dikarenakan arsitekturnya didesain sesuai dengan permasalahan yang ada. Jadi, jika permasalahan dirubah, maka sistem-sistem tersebut akan kesulitan untuk beradaptasi, dikarenakan solusi yang didapat tidak tepat atau kurang efisien. Algoritma genetika dibentuk untuk mengatasi permasalahan ini (Sastry, 2004). Sebuah algoritma genetika berfungsi mula-mula dengan menghasilkan himpunan dari solusi-solusi yang mungkin untuk masalah yang ada. Kemudian dilakukan evaluasi pada masing-masing solusi dan menentukan tingkat *Fitness* (ketahanan) untuk setiap himpunan solusi. Solusi-solusi tersebut kemudian menghasilkan solusi-solusi yang baru. Solusi-solusi *parent* yang lebih *fit* adalah yang memiliki kemungkinan besar untuk reproduksi, sedangkan yang kurang memiliki kemungkinan kecil untuk reproduksi. Pada intinya, solusi-solusi berevolusi dari waktu ke waktu. Dengan cara ini, dapat dikembangkan skop ruang pencarian pada suatu titik di mana bisa didapatkan solusi. Algoritma genetika bisa sangat efisien bila diprogram secara tepat.

3. POSI FORMULATION

Jika dalam datu database dijumpai sejumlah j dokumen/paper dimana setiap dokumen/paper memiliki kata kunci (k) terhadap I dimana I,j adalah integer, maka perhitungan untuk kemiripan antara sejumlah kata kunci (*keyword*) tersebut dapat dihitung dengan POSI Formulation. Misalkan dokumen/paper$_1$ disebut sebagai dokumen$_1$, paper $_2$ disebut sebagai dokumen$_2$ sampai dengan dokumen$_j$ disebut dengan dokumen$_j$.
Kromosom (kata kunci)$_1$ disebut dengan k_1, kromosom$_2$ disebut dengan K_2 dengan kromosom$_i$ disebut dengan K_i Untuk menguji persentasi kemiripan antara kata kunci (*keyword*) terhadap dokumen dapat dihitung dengan menggunakan perhitungan *Percentage of Similarity* (***POSI)*** *formulation.* Proses yang dilakukan adalah bahwa proses GA telah menghasilkan kata kunci solusi. Kemudian kata kunci ini akan dibandingkan dengan data yang ada pada database pada kolom judul tulisan, kata kunci keseluruhan pada tiap record, abstrak dan pada referensi. (*Sihombing, 2010*)

Formula yang digunakan dapat dilihat seperti pada formula 2.3 berikut ini.

$$\mathbf{Sim}\ (k,d) = \frac{\sum_{n} k_i\, d_j}{\mathbf{K}_{\mathbf{total}}} \qquad \text{.....(2.3)}$$

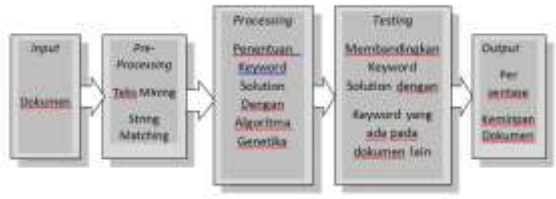
Dimana Sim (k,d) = Nilai Kemiripan.
$k_i d_j$ = jumlah masing-masing nilai kata kunci (i dan j = 0, 1,2,3,,,n adalah bilangan integer)

K_{total}= jumlah total dari semua kata kunci solusi yang terdiri dari judul, abstrak dan kata kunci.

4. METODOLOGI PENELITIAN

4.1 Skema Proses dan Aliran Data

Cara kerja pengukuran kemiripan dokumen ini adalah dimulai dengan pengumpulan data sesuai dengan kebutuhan penelitian, kemudian akan dilakukan *prepossessing* (*text mining*) untuk menghindari data yang tidak valid dalam pengujian. Selanjutnya akan dilakukan proses input data kedalam database sesuai dengan format yang telah ditentukan. Proses selanjuntnya adalah pemilihan dokumen untuk diuji dan melakukan pencocokan kata (*string matching)* untuk menghindari kata kunci (*keywords*) ganda pada setiap dokumen. Kemudian akan masuk ke prosesing pencarian kata kunci terbaik (*keyword Solution*) dengan melakukan kompetisi dengan algoritma genetika. Untuk mendapatkan hasil kemiripan dokumen dilakukan perhitungan. Skema aliran data dapat dilihat pada Gambar 1 berikut ini.



Gambar 1 Skema Proses Global dan Aliran Data

4.2 Jenis dan Sumber Data

Jenis data yang digunakan dalam penelitian ini adalah data berupa jurnal yang sudah terpublikasi dengan tipe file adalah Portable Doument File (PDF). Dengan rincian terdiri dari:

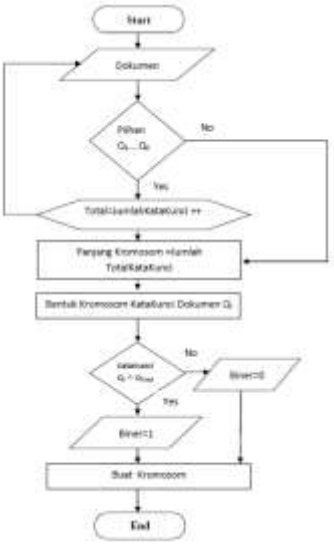
a. Kode dokumen dalam bentuk angka (*id*),
b. Judul penelitian (*title*),
c. Penulis, (*author*)
d. Nama Journal, Prosiding, Majalah Ilmiah, Volume dan Tahun
e. Abstrak,
f. Kata kunci (*keyword*),
g. Referensi (*Bibliography*)

Adapun sumber data jurnal diperoleh dari Prosiding Seminar Nasional Ilmu Komputer (SNIKOM) APTIKOM Wilayah I.

4.3 Pembentukan Kromosom Kata Kunci

Dengan n query dimana setiap dokumen Q_i (i = 0,1,..., n) memiliki kata kunci, kemudian seluruh kata kunci terpilih akan digabungkan. Kemudian akan dilakukan seleksi pada Q_{total} jika ada kata kunci yang sama maka akan

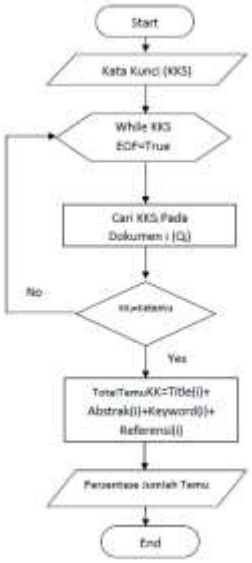
dipilih satu sehingga semua kata kunci tidak ada yang sama. Dari jumlah total kata kunci inilah yang menjadi panjang kromosom yang akan terbentuk. Dapat dijelaskan pada gambar 3.4 untuk menghasilkan kromosom kata kunci dari seluruh dokumen.



Gambar 2: Proses Pembentukan Kromosom

4.4 Metode Pengujian

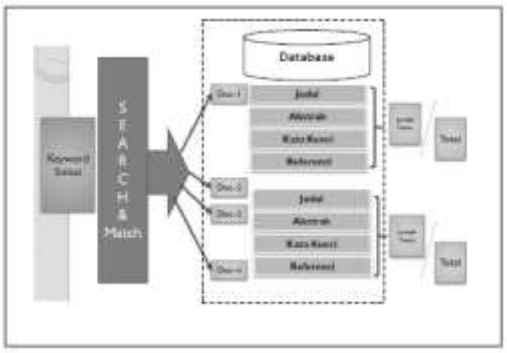
Adapun proses pengujian kemiripan dokumen adalah seperti gambar 3.1 berikut ini:



Gambar 3 Proses Pengujian Kemiripan

4.5 Perhitungan Kemiripan

Tahapan yang dilakukan adalah dengan membuat query dalam menyeleksi dokumen yang akan dibandingkan dengan database jurnal. Kemudian akan menghasilkan kata kunci solusi. Akan dilanjutkan dengan pembandingan dengan dokumen yang ada pada database dengan menggunakan query pada judul tulisan, kata kunci, abstrak, dan referensi sebuah dokumen. Pada database telah dilakukan pengelompokan berdasarkan kriteria seperti kategori jurnal/paper, tahun dan lainnya. Kemudian pengujian dilakukan dengan *Percentage of Similarity* (POSI Formulation) dan membuat laporan yang sudah dirangking berdasarkan nilai persentase kemiripan yang tertinggi (Sihombing, 2010). Adapun proses perhitungan yang dilakukan seperti pada Gambar 4 berikut ini.



Gambar 4. Perhitungan persentase kemiripan

Setelah dilakukan pencarian pada database dan jumlah kata kunci yang terdapat pada setiap dokumen akan dibagikan dengan total semua kata kunci yang ditemukan pada pada database. Proses perhitungan seperti berikut ini;

For i = 1 to Jumlah_Doc_Temu
Persentase Mirip Doc(i) = TotalTemuKKS_Doc(i)/ TotalTemuKKS All
Next i

Nilai persentase yang diperoleh dalam setiap pengujian akan berubah karena nilai fitness yang didapat pada proses algoritma genetika bisa berbeda. Perbedaan dapat meningkat keseluruhan ataupun akan berkurang, namun tidak merubah posisi atau tingkatan urutan kemiripan. Hal ini terjadi karena nilai random diberlakukan pada saat proses algoritma genetika.

5. PEMBAHASAN

Seperti yang telah dijelaskan dalam bab sebelumnya tujuan utama dari penelitian ini adalah mengetahui hasil persentase kemiripan sebuah dokumen karya ilmiah dengan dokumen karya ilmiah lainnya. Informasi yang didapat dari beberapa dokumen yang memiliki kategori yang sama, sehingga berdasarkan pengujian kemiripan dapat memberikan informasi lebih awal tentang isi dokumen tersebut. Untuk itu yang menjadi pembanding yang digunakan adalah kata kunci yang terdapat pada abstrak setiap tulisan yang ada. Aspek kedua yang akan dibahas adalah masalah peringkat kemiripan dokumen. Tujuan utama adalah untuk memilih kata kunci yang terbaik dari kata kunci yang dipilih oleh pengguna (*user*).

Dengan banyaknya tulisan/jurnal saat ini terutama dalam bentuk digital semakin memudahkan penulis mencari referensi. Namun pada kenyataannya banyak sekali dokumen yang mirip dalam kasus yang sama, termasuk referensi yang digunakan. Bagi pembaca atau user yang akan merujuk atau untuk mengetahui isi dari sebuah tulisan tentunya akan membutuhkan waktu yang banyak menyelesaikan itu. Sehingga perlu dilakukan pengujian kemiripan dokumen dengan menggunakan kata kunci yang ada ada setiap tulisan. Pada kenyataannya banyak sekali dokumen yang memiliki kata kunci yang mirip atau yang sama.

Dengan melihat masalah di atas bahwa sangat penting untuk menemukan kata kunci terbaik diantara semua kata kunci yang dipilih untuk dilakukan pengujian, sehingga diperlukan kompetisi kata kunci (*keyword competition*). Dimana kompetisi kata kunci ini merupakan proses untuk menyeleksi kata kunci yang

memiliki peluang kecil untuk diuji, maka proses ini akan menghasilkan Kata Kunci Solusi (*Keyword Solution*). Pada penjelasan berikutnya adalah tentang penyajian gambaran skema kompetisi kata kunci, kemudian diikuti oleh penjelasan pemilihan rinci komponen kompetisi kata kunci, menyajikan Kromosom kromosom, menyajikan Kromosom populasi awal, menyajikan evaluasi fungsi fitness, menyajikan proses seleksi, menyajikan proses crossover, dan proses mutasi, menyajikan proses rekombinasi dan menyajikan solusi kunci .

5.1 Memilih Dokumen Untuk Diuji

Pada pembentukan kromosom telah dijelaskan dengan memilih dokumen kemudian menggabungkan kata kunci (Q_{total}) kemudian akan menghilangkan kata kunci yang sama (duplikat) dengan cara membandingkan kata kunci Q_1, Q2,,,, Qn.

Oleh karena itu panjang kromosom diwakili oleh semua kata kunci dalam Q_{total}, dimana :

$\mathbf{Q}_{total} = \{Q_{0U}\ Q_{1U}\ Q_{2U}\ ....\ Q_{nU}\}$ ……...(4.1)

($\mathbf{Q}_{total}$ = satu set semua kata kunci)

Dengan mengambil contoh sebagai berikut: pilihan user terdiri dari lima (5) dokumen yang disebut Dokumen-0 (atau DOC-0), Dokumen -1 (atau DOC 1), Dokumen -2 (atau DOC-2), Dokumen-3 (atau DOC-3), dan Dokumen-4 (atau DOC-4), yang diwakili oleh kata kunci seperti yang ditunjukkan dalam tabel 1.

Tabel 1: Kata kunci Dari dokumen yang akan diuji

Query	Kata Kunci
Doc-0 (Q_0)	*watermarking, ssw, pseudo noise, watermark pattern,watermark recovery correlation, kriptografi*
Doc-1 (Q_1)	*citra digital, steganografi, least significant bit, peak signal noise ratio, kriptografi*
Doc-2 (Q_2)	*otp, playfair, kriptografi, kombinasi,* **citra digital**
Doc-3 (Q_3)	**kontras, brightness, grayscale, thresholding, inversi, citra biner, filter, noise, deteksi tepi, citra digital**
Doc-4 (Q_4)	*watermarking, looping, robustnes, dokumen gambar, least significant bit, kriptografi*
User's keywords (Q_{total})	*Jumlah Kata Kunci adalah 25 terdiri dari: watermarking, ssw, pseudo noise, watermark pattern,watermark recovery correlation, kriptografi, citra digital, steganografi, least significant bit, peak signal noise ratio, otp, playfair, kombinasi, kontras, brightness, grayscale, thresholding, inversi, citra biner, filter, noise, deteksi tepi, looping, robustness, dokumen gambar*

Dari Tabel 1 diperoleh:

Q_0 = (*watermarking, ssw, pseudo noise, watermark pattern,watermark recovery correlation, kriptografi*)

Q_1= (*citra digital, steganografi, least significant bit, peak signal noise ratio, kriptografi*)

Q_2= (*otp, playfair, kriptografi, kombinasi*, **citra digital**)

Q_3= (**kontras, brightness, grayscale, thresholding, inversi, citra biner, filter, noise, deteksi tepi, citra digital)**

Q4= (watermarking, looping, robustnes, dokumen gambar, least significant bit, kriptografi)

Oleh karena itu, dengan Persamaan 4.1 jumlah panjang kromosom dari semua dokumen diwakili oleh Q_{total} adalah $\{Q_{0U}\ Q_{1U}\ Q_{2U}\ Q_{3U}\ Q_{4U}\}$, sehingga mendapatkan Q_{total} sebagai berikut:

Q_{total}= (*watermarking, ssw, pseudo noise, watermark pattern,watermark recovery correlation, kriptografi, citra digital, steganografi, least significant bit, peak signal noise ratio, otp, playfair, kombinasi, kontras, brightness, grayscale, thresholding, inversi, citra biner, filter, noise, deteksi tepi, looping, robustness, dokumen gambar*)

Total elemen dalam Q_{total} adalah 25 kata kunci, karena satu kata kunci hanya ditampilkan satu kali. Dengan kata lain dengan Tabel 1 di atas, memiliki lima (5) dokumen sebagai masukan dimana setiap dokumen memiliki kata kunci. Gabungan kata kunci akan menentukan kromosom.

Oleh karena itu, dengan persamaan (1) Q_{total} mendapatkan jumlah semua panjang kromosom adalah $|Q_{total}|$ = 25 bit sebagai berikut:

Jumlah Kata kunci user (Q_{total})	*Jumlah Kata kunci = 25: watermarking, ssw, pseudo noise, watermark pattern,watermark recovery correlation, kriptografi, citra digital, steganografi, least significant bit, peak signal noise ratio, otp, playfair, kombinasi, kontras, brightness, grayscale, thresholding, inversi, citra biner, filter, noise, deteksi tepi, looping, robustness, dokumen gambar*

5.2 Representasi Kata Kunci

Kata kunci diwakili oleh kromosom. Sebuah kromosom merupakan dokumen tunggal yang mengandung kata kunci. Dalam penelitian ini pertanyaan yang diberikan n pengguna (Q_n) dengan masing-masing query memiliki kata kunci. Kemudian mendefinisikan populasi awal sebagai kromosom n dan jumlah semua panjang kromosom adalah Q_{total}. Sebuah kromosom adalah serangkaian kata kunci yang diwakili oleh bit (0 dan 1). Ketika kata kunci disajikan dalam dokumen, bit diatur ke 1; jika tidak diatur ke 0. Setiap dokumen kemudian dapat diwakili dalam urutan 0 dan 1, yang disebut model kromosom.

Kromosom (Contoh1)

10000000000000000000000**1**

Artinya kromosom pada kata kunci terdiri dari :
watermarking and ***dokumen gambar.***

Dari contoh kata kunci diatas dapat digambarkan kromosom berikut ini:

Kromosom (contoh2)

11000000000000000000000000

Artinya kromosom pada kata kunci terdiri dari : ***watermarking*** *dan* ***ssw***

Kata Kunci ***watermarking*** ,***ssw*** ,00000000000000000000000

(contoh)

Menggunakan Tabel 4.1 query total = 25 bit dengan populasi awal dari 5 dokumen sebagai berikut: Q_{total} = {watermarking, ssw, pseudo noise, watermark pattern,watermark recovery correlation,citra digital, steganografi, least significant bit, peak signal noise ratio, kriptografi, otp, playfair, kombinasi, kontras, brightness, grayscale, thresholding, inversi, citra biner, filter, noise, deteksi tepi, looping, robustness, dokumen gambar}.

Berikut ini adalah kata kunci dari Doc-0 sebgai berikut :

Query	Kata Kunci
DOC-0 (Q_0)	*watermarking, ssw, pseudo noise, watermark pattern, watermark recovery correlation, kriptografi*

kemudian sesuai dengan kata kunci dari Doc-0 (Q_0) dengan kata kunci Q_{total}. Jika kata kunci dari Doc-0 (Q_0) ada di kata kunci dari Q_{total}, bit diubah menjadi nilai 1; jika tidak diberikan nilai 0. Sehingga menghasilkan Doc-0 sebagai berikut:

Kromosom
11111100000000000000000000
dari Doc-0 (C_0)

Dengan cara yang sama seperti diatas, jika dilihat kata kunci dari Doc-1 sebagai berikut:

Query	Kata Kunci
Doc-1 (Q_1)	*citra digital, steganografi, least significant bit, peak signal noise ratio, kriptografi*

Oleh karena itu kromosom Doc-1 sebagai berikut:

Kromosom
00000111110000000000000000
dari Doc-1 (C_1)

Kata kunci pada dokumen 2:

Query	Kata Kunci
DOC-2 (Q_2)	*otp, playfair, kriptografi, kombinasi,* **citra digital**

Maka kromosom dari dokumen 2 adalah :

Kromosom
00000100011110000000000

dari Doc-2 (C_2)

Kata kunci pada dokumen 3:

Query	Kata Kunci
Doc-3 (Q_3)	**kontras, brightness, grayscale, thresholding, inversi, citra biner, filter, noise, deteksi tepi, citra digital**

Sehingga kromosom dari dokumen 3 adalah

Kromosom
00000100000011111111100

dari Doc-3 (C_3)

Berikut adalah kata kunci pada dokumen 4:

Query	Kata Kunci
Doc-4 (Q_4)	*watermarking, looping, robustnes, dokumen gambar, least significant bit, kriptografi*

Sehingga diperoleh kromosom dari dukumen 4 adalah:

Kromosom
10000001010000000000000111

dari Doc-4 (C_4)

Kromosom kata kunci dari Doc-0, Doc-1, Doc-2, Doc-3, Doc-4, adalah sebagai berikut:
• Kromosom Doc-0 (C_0) = 11111100000000000000000000
• Kromosom Doc-1 (C_1) = 00000111110000000000000000
• Kromosom Doc-2 (C_2) = 00000100011110000000000000
• Kromosom Doc-3 (C_3) = 00000100000000011111111100
• Kromosom Doc-4 (C_4) = 10000001010000000000000111

5.3 Evaluasi *Fitness* Kata Kunci

Setelah mendapatkan Kromosom pertama, maka sistem akan menentukan kata kunci sebagai kromosom pada Kromosom berikutnya. Fitness kata kunci adalah nilai yang digunakan untuk memilih kata kunci pada Kromosom berikutnya. Ini adalah salah satu operator yang paling penting dalam proses kompetisin kata kunci. Fungsi ini memetakan semua sifat-sifat satu individu untuk, pada dasarnya memberikan peringkat dan tempat di antara individu-individu lain. Menciptakan operator *fitness* adalah salah satu tugas paling sulit dalam menciptakan skema kompetisi kata kunci. Hal ini dibutuhkan aga dapat mengambil semua contoh dan mempertimbangkan individu sebagai solusi. Kemudian akan dilanjutkan pada pemetaan beberapa kekurangan dari solusi.
Tujuan dari fase ini adalah untuk mendapatkan nilai fitness dari kata kunci. Fitness kata kunci adalah nilai yang digunakan dalam memilih setiap kata kunci untuk Kromosom berikutnya. Evaluasi nilai *fitness* tergantung pada kasus setiap kata kunci dalam setiap Kromosom. Dalam penulisan ini nilai fitness akan menentukan kemiripan antara kata kunci dan dokumen. Dalam tulisan ini menggunakan fungsi Jaccard dan fungsi *Cosine* untuk mencari nilai fitness. Karena fungsi ini salah satu metode yang baik untuk melakukan pengujian kemiripan.
Skema ini akan mengevaluasi berdasarkan fungsi Jaccard dan fungsi Cosine sebagai formulasi.

Untuk pengujian data menggunakan data set dari " Prosiding SNIKOM". Data yang digunakan adalah koleksi sejumlah dokumen dengan struktur sebagai berikut: Id-doc (nomor jurnal), Judul (JDL), Penulis (PNL), Abstrak (ABS), Kata Kunci (KWD) dan Referensi (REF).

5.4 Proses Pemilihan Kata Kunci

Proses pemilihan kata kunci adalah sama dengan proses seleksi alam dalam evolusi. Prinsip proses pemilihan kata kunci adalah sebagai berikut: the "*mostfitness*" anggota populasi kunci akan bertahan, dan "fitnesss lemah" akan dieliminasi. Proses pemilihan kata kunci adalah algoritma yang algoritma evolusi menuju solusi yang terbaik. Dalam penelitian ini, penentuan populasi kunci pada Kromosom berikutnya dipilih berdasarkan nilai fitness kata kunci. Semakin tinggi tingkat nilai fitness maka probabilitas untuk dilanjutkan sangat besar, sementara yang memiliki nilai rendah akan ditinggalkan atau dihapus.

Proses seleksi didasarkan pada seleksi roda roulette. Yaitu dengan memutar roda n kali, pada setiap pemilihan kata kunci. Dengan cara ini, kata kunci yang terbaik secara alami akan menghasilkan nilai rata-rata yang tinggi, dan kata kunci terburuk akan mendapatkan nilai yang rendah.
Proses pemilihan kata kunci adalah langkah untuk memilih dua kromosom kata kunci dari populasi kunci sesuai dengan fitness (*fitness yang lebih baik, kesempatan lebih besar untuk dipilih*). Induk yang dipilih sesuai dengan fitness kata kunci yang ada.
Agar lebih jelas, dapat dilihat kromosom kata kunci sebelumnya, mengatakan dengan kromosom A, B, C, D, dan E sebagai berikut.
Nilai fitness dari setiap kata kunci adalah sebagai berikut:
C_0 = 11111100000000000000000000→ *fitness* C_0 dari Q_0 = 0,25
C_1 = 00000111110000000000000000→ *fitness* C1 dari Q_1 = 0,27
C_2 = 00000100011110000000000000→ *fitness* C2 dari Q_2 = 0,26
C_3 = 00000100000000011111111100→ *fitness* C3 dari Q_3 = 0,24
C_4 = 10000001010000000000000111→ *fitness* C4 dari Q_4 = 0,28
Total nilai fitness = 0.25 + 0.26 + 0.27 + 0.24 + 0.28 = 1,3. Oleh karena itu, dapat ditunjukkan dalam diagram berikut:

• 0.25 / 1,3 * 100% = 19% (wilayah A)
• 0.26 / 1,3 * 100% = 20% (wilayah B)
• 0.27 / 1,3 * 100% = 21% (wilayah C)
• 0.24 / 1,3 * 100% = 18% (wilayah D)
• 0.28 / 1,3 * 100% = 22% menyebutnya wilayah E)

Dalam proses selanjutnya akan memproses kromosom kata kunci ini lagi sampai pada nilai fitness kata kunci yang terbaik, berarti nilai yang ditemukan sudah stabil sebagai kata kunci solusi. Dari contoh di atas dapat dilihat nilai fitness kata kunci yang terbaik dari semua kata kunci pada nilai 0,28. Dengan kromosom kata kunci sebagai berikut:

10000001010000000000000111**.**

5.5 *Crossover* Kromosom Kata Kunci

Crossover kromosom kata kunci adalah kemungkinan melakukan pertukaran antar populasi. Dari sebuah persilangan akan menyeberang dari satu populasi ke populasi lainnya untuk membentuk keturunan baru (anak). Jika tidak ada *Crossover* dilakukan, maka secara otomatis akan mengambil data induk (*parent*). Ini hanyalah pertukaran bit dari 0 menjadi 1 dan dari 1 menjadi 0. Nilai yang paling baik adalah ≥ 0,7. Mengapa? Nilai ini adalah ukuran umum dalam Crossover, jika memilih 0,6 atau 0,5 maka nilai tersebut terlalu kecil. Oleh karena itu sebaiknya mengambil nilai yang terbaik adalah sekitar ≥ 0,7 (dari skala 0,1 sampai 0,9). *Crossover* dilakukan dengan memilih Kromosom acak sepanjang kromosom dan swapping semua Kromosom.

Agar lebih jelas, dapat dilihat pada penjelasan berikut ini :

Kata kunci Kromosom 1: *100010011* | *10010010*
Kata kunci Kromosom 2: 010100010 | 01000011

5.6 Mutasi Kromosom Kata Kunci

Tujuan dari mutasi adalah untuk menghasilkan kata kunci dengan keturunan baru pada setiap posisi dalam kromosom. Mutasi adalah kesempatan dalam kromosom untuk membalik (0 menjadi 1, 1 menjadi 0). Mutasi dilakukan dengan ditentukan nilai yang. Nilai konstan biasanya ditetapkan dengan nilai sangat rendah, misalnya 0,01. Setiap kali kromosom dipilih dari populasi algoritma, pertama sekali memeriksa dan melihat apakah Crossover harus diterapkan dan kemudian perulangan dilakukan disepanjang masing-masing kromosom yang bermutasi. Iterasi dilakukan untuk setiap kromosom dengan mengambil nilai acak dari 0 sampai 1. Jika nilai acak yang dihasilkan, maka kromosom/bit terbalik, jika tidak ada yang dilakukan. Mutasi ini dimaksudkan untuk mencegah berkurangnya dari semua solusi dalam populasi sehingga dapat menjadi optimum sehingga masalah dapat diselesaikan. Proses mutasi acak dapat mengubah keturunan yang dihasilkan dari crossover. Dalam kasus pengkodean biner dapat bertukar beberapa bit yang dipilih secara acak 1-0 atau dari 0 sampai 1.
Contoh mutasi ditunjukkan dalam Tabel 2.

Tabel 2: Mutasi kromosom kata kunci

Keturunan Asli 1	110**1**111000011110
Keturunan Asli 2	110110***0***100110***1***0
Keturunan Bermutasi 1	110**0**111000011110
Keturunan Bermutasi 2	110110***1***100110***0***0

Pada tabel 2, dapat dilihat teknik mutasi (*serta crossover*) terutama tergantung pada pengaturan dari kromosom. Dari contoh di atas, mutasi bisa dilakukan sebagai pertukaran dua Kromosom.

6.7 Kromosom Kata Kunci Solusi (*Kromosom Terakhir*)

Pada kromosom contoh sebelumnya, setelah beberapa kromosom ditemukan kromosom tunggal dari semua populasi dengan nilai fitness yang memiliki nilai konstan: 0,28, sebagai berikut:

10000001010000000000000111
[0,28]

Oleh karena itu, sistem akan melaporkan bahwa solusi kata kunci dalam Kromosom terakhir adalah sebagai berikut:

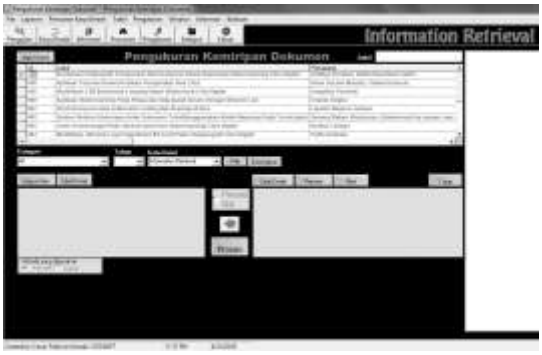
10000001010000000000000111

melihat kromosom contoh di atas, ada 6 kata kunci yang muncul sesuai dengan urutan nomor satu (1). Posisi pertama di 11 (bit pertama) adalah 1, maka kata kunci "watermaking", nomor 2 pada posisi 8 (bit kedelapan) adalah 1, maka kata kunci "least significant bit", nomor 3 pada posisi bit 10 adalah 1, maka kata kunci "kriptografi", nomor 4 pada posisi bit ke 22 adalah 1, maka kata kunci "deteksi tepi", nomor 5 pada posisi bit ke 23 adalah 1, maka kata kunci "looping", dan nomor ke 6 pada posisi 25, dengan kata kunci "dokumen gambar". Ini berarti bahwa kata kunci bernilai 1 berarti dipilih, dan nilai 0 tidak akan dipilih. Jadi kata kunci solusi yang ditemukan adalah: watermaking, least significant bit, kriptografi, deteksi tepi, looping, dokumen gambar, maka sesuai dengan kata kunci tersebut, sistem akan menyarankan dokumen database yang mirip dengan pilihan user.

6. PENGUJIAN DAN HASIL

6.1 Prototipe Aplikasi

Dalam melakukan pengujian menggunakan aplikasi yang dirancang dalam memudahkan menentukan pilihan dokumen untuk dilakukan pengujian kemiripan. Tampilan seperti berikut ini.
Langkah 1: Tampilkan semua dokumen seperti pada Gambar 5 berikut ini :



Gambar 5 Tampilan daftar dokumen

Langkah 2: Lakukan pemilihan dokumen dari tabel dengan klik pada judul dokumen atau dapat melakukan seleksi dengan memilih kategori, tahun dan kata kunci, dengan memilih IdDoc-664, IdDoc-659, IdDoc-660, dan IdDoc-657. Seperti pada gambar 6 berikut ini:

Gambar 6 Tampilan pemilihan dokumen yang akan diuji

Langkah 3: berikan tanda ceklist pada GA Report dan kemudian Klik tombol Proses dan akan menghasilkan laporan proses pengujian dan persentase kemiripan dokumen seperti pada gambar 7.

Gambar 7 Laporan Proses Pengujian dan Kemiripan

6.2 Pengujian Kemiripan Dokumen.

Berikut ini akan dijelaskan hasil pengujian dengan menggunakan aplikasi yang dirancang. Pada proses ini dengan memilih 4 dokumen yaitu dokumen dengan IdDoc-664, IdDoc-659, IdDoc-660, dan IdDoc-657 yang ditampilkan pada tabel 3.

Tabel 3 Sumber dokumen yang diuji.

No	ID-Doc	Judul Dokumen	Kata Kunci
1	664	Modifikasi Metode Least Significant Bit (Lsb) Pada Steganografi Citra Digital	Kriptografi, Least Significant Bit, Steganografi, Citra
2	659	Modifikasi LSB Berbentuk Looping Dalam Watermark Citra Digital	watermarking, Least Significant Bit, looping, robustness
3	660	Aplikasi Turunan Numerik Dalam Pengenalan Pola Citra	brightness, citra biner, deteksi tepi, filter, grayscale, inversi, noise, thresholding, Least Significant Bit, kontras
4	657	Kombinasi Kriptografi Transposisi Dan Kompresi Untuk Keamanan Watermarking Citra Digital	kompresi, kompresi shannon-fano, kriptografi transposisi, Least Significant Bit, transposisi segitiga, transposisi segitiga, transposisi spiral, transposisi zig-zag, watermarking

Pada tabel 3 ini ditampilkan hasil pengujian dengan seluruh dokumen yang ada dalam database dengan menggunakan query untuk pengujiannya. Persentase kemiripan dari empat dokumen terpilih memperoleh kemiripan dengan persentase tertinggi adalah 32,26%, peringkat kedua 24,19%, peringkat ketiga 19,35%, peringkat keempat 12,90% dan seterusnya pada peringkat terakhir kemiripan dengan 11,19%. Nilai persentase kemiripan ini akan berubah tergantung hasil pemilihan kata kunci solusi.

Pada uji coba yang dilakukan dalam penelitian ini menghasilkan seperti pada tabel 4 berikut ini.

Tabel 4. Pengujian dan Hasil

No	Id-Doc	Kemiripan	Judul Jurnal
1.	661	32.26%	Studi Kompresi Data Arithmetic Coding Dan Kriptografi RSA
2.	665	24.19%	Pengamanan pesan teks dengan kombinasi algoritma kriptografi one-time pad dan playfair cipher
3.	663	19.35%	Jenis Penyerangan Pada Spread Spectrum Watermarking Citra Digital
4.	667	12.90%	Meningkatkan Robustness Watermarking Audio Digital Melalui MSB Dan Algortima RSA
5.	666	11.29%	Pengenalan Tanda Tangan Digital Dengan Menggunakan Metode Learning Vector Quantization (LVQ)

7.3 Perhitungan Persentase Kemiripan.

Berikut ini tabel proses pengujian dan perhitungan kata kata kunci solusi dapat ditunjukkan pada tabel 5 berikut ini.

Tabel 5 Perhitungan Kata kunci solusi

Kromosom Kata Kunci	Kata Kunci Solusi	Jumlah Kata Kunci Dalam Dokumen				
		Doc-661	Doc-665	Doc-663	Doc-667	Doc-666
110011 110010 101000 00000	Kriptografi	2	3	2	1	1
	Least Significant Bit	5	2	2	2	2
	Watermarking	4	4	2	1	0
	Looping	2	1	2	0	1
	Robustness	2	1	0	1	0
	Deteksi tepi	2	1	1	1	1
	Grayscale	2	1	2	1	2
	Noise	1	2	1	1	0
	Total Kata Kunci	20	15	12	8	7
Persentase Kemiripan (%)		32.26	24.19	19.35	12.90	11.29

Dari tabel di atas ditampilan hanya 5 dokumen yang memiliki nilai lebih besar. Ber dasarkan pengujian diatas maka kata kunci solusi yang dihasilkan dibandingkan dengan dokumen doc-661 terdapat 20 kata yang terdiri dari kriptografi ditemukan 2, untuk kata Least Significant Bit ditemukan 5, kata Watermarking ditemukan 4, kata Looping ditemukan 2, kata Robustness ditemukan 2, kata Deteksi tepi ditemukan 2, kata Grayscale ditemukan 2, sedangkan kata Noise ditemukan 1 kata kunci. Sehingga total 2+5+4+2+2+2+2+1=20, dengan cara yang sama dilakukan maka jumlah kata kunci yang ditemukan pada doc-665 = 15 , pada doc-663 ditemukan 12, pada doc-667=8 sedangkan doc-666 ditemukan jumlah kata 7.

Sehingga total kata yang ditemukan pada keseluruhan data yang ada pada database 20+15+12+8+7=62 kata.
Persentase kemiripan dari hasil pengukuran antara lain:

- Persentase Kemiripan doc-661 adalah 20/81 = 32,26 %
- Persentase Kemiripan doc-665 adalah 15/81 = 24,19 %
- Persentase Kemiripan doc-663 adalah 12/81 = 19,35 %
- Persentase Kemiripan doc-667 adalah 8/81 = 12,90 %
- Persentase Kemiripan doc-666 adalah 7/81 = 11,29 %

Sesuai melalui proses perhitungan diatas maka persentase kemiripan yang paling besar antara dokumen (doc-664, doc-659, do-658, doc-657) adalah pada dokumen-661. Pada contoh pengujian di atas jumlah batas dokumen yang diuji kemiripannya sejumlah 8 namun yang disajikan adalah 5 dokumen yang memiliki nilai persentase lebih tinggi.

7. PENUTUP

Setelah membahas dan menguji kemiripan dokumen maka dapat diberikan beberapa kesimpulan antara lain:

1. Pengukuran kemiripan dokumen karya ilmiah dapat dilakukan dengan membandingkan jumlah kata kunci yang terdapat dalam sebuah dokumen karya ilmiah dengan dokumen karya ilmiah lainnya dalam waktu yang lebih cepat (*satuan waktu detik*) dibandingkan pada pengujian dengan membaca isi dokumen karya ilmiah secara manual.
2. Dari pengujian data yang dilakukan diperoleh hasil kemiripan dari dokumen IdDoc-661 sebesar 32,26%, urutan kedua 24,19% pada IdDoc-665 dan urutan ketiga sejumlah 19,35% pada IdDoc-663 dan urutan keempat sebesar 12,90% pada IdDoc-667 dan yang terakhir sebesar 11,29% pada IdDoc-666.
3. Aplikasi yang digunakan merupakan salah satu tools yang dapat menjalankan proses dengan baik yang menghasilkan kunci solusi untuk melakukan pengujian.

Untuk kesempurnaan dari penelitian ini diberikan beberapa saran antara lain:

1. Mengingat semakin banyak dokumen yang dipilih untuk dibandingkan maka akan membuat jumlah kromosom semakin panjang dan waktu yang lebih banyak dan mengakibatkan proses optimisasi berjalan lambat. Oleh karena itu, disarankan agar terdapat suatu metode tertentu yang digunakan dalam pemilihan kata kunci-kata kunci tertentu yang akan digunakan dalam model kromosom.
2. Aplikasi yang sudah dirancang digunakan secara multiuser dalam satu jaringan Local Area Network, sebaiknya dikembangkan untuk online dengan berbasis web.

DAFTAR PUSTAKA

Baeza-Yates, R, Ribeiro-Neto, B. 1999 Modem Information Retrieval. Addison Wesley.

Basuki, A. Algoritma Genetika, 2003 Suatu Alternatif Penyelesaian Permasalahan Searching, Optimasi dan Machine Learning, PENS-ITS, Surabaya

Berthon, P.,N.C a.J Hulbert et al., 2007. " Organizational and Customer Prespective on Brand Equity: Issues for Manager and Researchers" Bentley College-Columbia University-Curtin university of Technology.

Candra Triawati, 2009, Institut Teknologi Telkom , "Metode Pembobotan Statistical Concept Based untuk Klastering dan Kategorisasi Dokumen Berbahasa Indonesia",diakses pada, http://www.ittelkom.ac.id/library/index.php?view=article&catid=20%3Ainformatika&id=590%3Atextmining&option=com_content&Itemid=15, TanggalAkses : 04 Desember2011

Even, Zohar, 2002 , Introduction To Text Mining. [online], Tersedia di: http://www.docstoc.com/docs/25443990/Introduction-to-Text-Mining diakses pada 16 Februari 2012

Gen, M. Dan Cheng R. 1997. Genetic Algoritm and Enginering design, Ashikaga Institute of Technology Ashikaga, Japan, A Wiley-Interscience publication, Jhon wiley & Sons, Inc

Gerald J. Kowalski, 2000. Information Storage and Retrieval Systems: Theory and Implementation, United States

Goldberg, D.E., 1989 Genetic Algorithm in Search, Optimization and Machine Learning. Addison-Weasley Publishing Company Inc, New York

Hearst, Marti. 2003. "What Is Text Mining." Retrieved October 18: 2005.

Hiemstra, Djoerd, 2009. Information Retrieval Models, Information Retrieval : Searching in 21st Century, Wiley

Hultberg, Jens dan Helger, Joakim Poromaa. 2007. Seminar course in Algorithms - Project report : String Matching

Kristanto, 2004, Jaringan Syaraf Tiruan (Konsep Dasar, Algoritma, dan Aplikasinya), Gava Media, Yogyakarta

Kusumadewi, S., 2003. Artificial Intelligence (Teknik dan Aplikasinya), Graha Ilmu, Yogyakarta

Mitchell, M. 1996 . an Introduction to Genetic Algorithms (2nd ed.). The MIT Press., London.

Niwattanakul, S., Singthongchai, J., Naenudorn, E., & Wanapu, S. 2013. Using of Jaccard Coefficient for Keywords Similarity. Proceedings of the International MultiConference of Engineers and Computer Scientists, Hongkong

Sastry, K. et.al. 2004, Genetic Programming for Multiscale Modeling. Urbana: University of Illinois at Urbana, Champaign

Sihombing Poltak, 2010,. Keyword Competition Approach in Ranked Document Retrieval, Disertasi, Universiti Sains Malaysia, Penang, Malaysia.

Suyanto. 2005. Algoritma Genetika dalam Matlab, : Andi., Yogyakarta

Tan, Ah-Hwee, 1999, Text Mining: The state of the art and the challenges, Kent Ridge Digital Labs 21 Heng Mui Keng Terrace Singapore 119613, online pada (http://www3.ntu.edu.sg/sce/labs/erlab/publications/papers/asahtan/ tm_pakdd99.pdf diakses pada tgl 19/11/2012)

Taufiq M. Isa, Taufik Fuadi Abidin, 2013. Mengukur Tingkat Kesamaan Paragraf Menggunakan Vector Space Model untuk Mendeteksi Plagiarisme, Seminar Nasional dan ExpoTeknik Elektro, Jurusan Matematika, FMIPA, Universitas Syiah Kuala.

Triawati, Chandra 2009, Metode Pembobotan Statistical Concept Based untuk Klastering dan Kategorisasi Dokumen Berbahasa Indonesia, Institut Teknologi Telkom Bandung.

Winoto, Hendri. 2012. Deteksi Kemiripan Isi Dokumen Teks Menggunakan Algoritma Levenshtein Distance. Teknik Informartika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim, Malang.