

ANALISIS PERBANDINGAN ALGORITMA LEVENSHEIN DISTANCE DAN JARO WINKLER UNTUK APLIKASI DETEKSI PLAGIARISME DOKUMEN TEKS

Michael Julian Tannga¹, Syaiful Rahman², Hasniati³

^{1,2,3}Teknik Informatika, STMIK Kharisma Makassar

Email: mik3_sturdy@yahoo.com, syaifulrahman@kharisma.ac.id, hasniati@kharisma.ac.id

ABSTRAK

Penelitian ini bertujuan untuk mengukur perbandingan kinerja algoritma Levenshtein Distance dan algoritma Jaro Winkler dalam mendeteksi plagiarisme dokumen teks. Data pengujian yang digunakan pada penelitian ini terdiri dari dua data uji, yaitu data uji untuk mengukur similarity algoritma dan data uji untuk mengukur waktu proses algoritma. Pengujian algoritma dilakukan dengan menggunakan aplikasi deteksi plagiarisme untuk menghitung nilai similarity dan waktu proses yang dihasilkan oleh kedua algoritma. Setelah dilakukan pengujian, data dari hasil kedua pengujian dihitung rata-ratanya kemudian dianalisis perbandingannya. Hasil analisis perbandingan yang didapat untuk rata-rata similarity dari algoritma Jaro Winkler yaitu sebesar 80.92 %, sedangkan untuk algoritma Levenshtein Distance rata-rata nilai similarity-nya yaitu sebesar 49.43%. Kemudian analisis perbandingan rata-rata waktu proses yang dimiliki algoritma Jaro Winkler yaitu 0.054 detik, sedangkan rata-rata waktu proses yang dimiliki algoritma Levenshtein Distance yaitu 0.138 detik. Berdasarkan perbandingan analisis yang telah dilakukan, algoritma Jaro Winkler terbukti memiliki akurasi similarity yang lebih tinggi dan waktu proses yang lebih cepat dibanding algoritma Levenshtein Distance dalam mendeteksi plagiarisme dokumen.

Kata kunci : Algoritma Jaro Winkler, Algoritma Levenshtein Distance, Plagiarisme

ABSTRACT

The goal of this study is to measure the performance comparison between Levenshtein Distance and Jaro Winkler algorithms to detect plagiarism in text documents. The test data that were being used in this study consisted of two test data, the test data to measure the similarity algorithms and test data to measure the processing time of the algorithms. The algorithm was tested by using plagiarism detection application to calculate the value of similarity and processing time by both algorithms. After testing, the results of the two tests are averaged and then analyzed the comparison. Results obtained for the comparative analysis of the average similarity of Jaro Winkler algorithm is 80.92%, while for the algorithm Levenshtein Distance is 49.43%. Then, comparative analysis of the average processing time of Jaro Winkler algorithm is 0.054 seconds, while the average processing time of Levenshtein Distance algorithm is 0.138 seconds. Based on the comparative analysis that has been done, Jaro Winkler algorithm is shown to have a higher similarity accuracy and the processing time is faster than the Levenshtein Distance algorithm in detecting plagiarism document.

Keywords: Jaro Winkler Algorithm, Levenshtein Distance Algorithm, Plagiarisme

PENDAHULUAN

Dalam dunia akademik, plagiarisme/penjiplakan cukup sering terjadi, khususnya di kalangan mahasiswa/pelajar. Hal ini disebabkan perkembangan teknologi yang membuat mahasiswa/pelajar selalu berinteraksi dengan komputer, mengingat adanya fasilitas untuk menyalin dan mengubah teks (copy and paste) sehingga mempermudah terjadinya tindakan plagiarisme.

Menurut Kamus Besar Bahasa Indonesia (KBBI), plagiarisme/penjiplakan berarti menggambar atau menulis garis-garis gambaran atau tulisan yang telah tersedia, mencontoh atau

meniru tulisan atau pekerjaan orang lain, mencuri karangan orang lain dan mengakui sebagai karangan sendiri atau mengutip karangan orang lain tanpa seizin penulisnya.

Metode pencocokan string merupakan salah satu metode yang banyak diterapkan untuk pencarian string berdasarkan kemiripan penulisan. Pada umumnya algoritma yang paling sering digunakan untuk pencarian/pencocokan string yaitu algoritma Levenshtein Distance dan Jaro-Winkler. Akan tetapi analisis perbandingan kinerja antara algoritma Levenshtein Distance dan Jaro-Winkler untuk deteksi plagiat dokumen belum pernah dilakukan. Oleh karena itu, penelitian ini bertujuan melakukan analisa terhadap perbandingan kinerja dari kedua algoritma tersebut sehingga dapat diketahui algoritma mana yang lebih akurat dan tepat dalam mendeteksi kesamaan dokumen. Algoritma Levenshtein Distance dan Algoritma Jaro Winkler merupakan algoritma pencocokan string yang tepat dalam mengukur kesamaan dari dua string atau dokumen.

Pada penelitian ini penulis mengukur perbandingan kinerja algoritma Levenshtein Distance dan Jaro Winkler untuk aplikasi deteksi plagiarisme dokumen teks.

Secara teoritis, hasil penelitian ini diharapkan dapat bermanfaat bagi institusi atau tenaga pengajar untuk mempermudah pengecekan terhadap hasil tulisan mahasiswa dengan membandingkan tulisan mahasiswa lainnya dengan waktu yang lebih cepat, ketelitian yang lebih tinggi dan usaha yang lebih kecil dibandingkan pendeteksian manual. Bagi penulis, mendapatkan pemahaman dan informasi mengenai algoritma mana yang tepat untuk digunakan dalam mendeteksi kesamaan dokumen. Bagi akademis, dapat menjadi rujukan bagi peneliti lain yang ingin melakukan penelitian lanjutan ataupun penelitian terkait.

LANDASAN TEORI

Algoritma *Levenshtein Distance*

Menurut Junedy (2014), Levenshtein Distance merupakan algoritma yang digunakan untuk mengukur nilai kesamaan atau kemiripan antara dua buah kata (string). Levenshtein Distance dibuat oleh Vladimir Levenshtein pada tahun 1965. Perhitungan Levenshtein Distance didapatkan dari matriks yang digunakan untuk menghitung jumlah perbedaan string antara dua string. Perhitungan jarak antara dua string ini ditentukan dari jumlah minimum operasi perubahan untuk membuat string A menjadi string B. Secara umum, operasi mengubah yang diperbolehkan untuk Algoritma ini ada 3 macam operasi, yaitu:

1. Insertion (penyisipan) : operasi penyisipan sebuah karakter kedalam string tertentu. Misalnya, penyisipan sebuah karakter „u“ kedalam string “bku” tepat setelah karakter „b”. Setelah operasi Insertion dilakukan, string “bku” berubah menjadi “bku”.
2. Deletion (penghapusan) : operasi penghilangan atau penghapusan sebuah karakter tertentu dari sebuah string. Misalnya menghapus karakter „s” pada string “senam”. Setelah operasi Deletion dilakukan, string berubah menjadi “enam”.
3. Substitution (penukaran): operasi penukaran sebuah karakter pada string tertentu dengan karakter lain. Misalnya, menukarkan karakter „s” pada string “sama” dengan karakter baru „l”. Setelah operasi substitution dilakukan, string akan menjadi “lama”.

Algoritma ini berjalan mulai dari pojok kiri atas sebuah array dua dimensi yang telah diisi sejumlah karakter string awal dan string target dan diberikan nilai cost. Nilai cost pada ujung kanan

bawah menjadi nilai levenshtein distance yang menggambarkan jumlah perbedaan antara dua string tersebut. Misalnya terdapat dua string yaitu string asli (CS) adalah aku dan string pembandingan (ST) adalah abu. Proses algoritma *levenshtein* sebagai berikut:

1. Jika panjang CS adalah 0 maka jarak CS ke ST adalah panjang ST dalam contoh diatas kata abu memiliki panjang atau bernilai 3 .
2. Jika panjang ST adalah 0 maka jarak CS ke ST adalah panjang CS dalam contoh diatas kata aku memiliki panjang atau bernilai 3 .
3. Membuat matriks dengan ukuran $(CS+1) \times (ST+1)$

Dalam penerapannya, pencocokan string dilakukan kalimat perkalimat. Kalimat dari proses sorting yang dalam bentuk array diubah dulu menjadi bentuk string, kemudian string tersebut akan dihitung nilai similarity-nya menggunakan rumus:

$$Plagiarized\ Value = \left\{ 1 - \frac{Diff}{\text{Max}(CS, ST)} \right\} \times 100\% \quad (1)$$

Keterangan :

Plagiarize value = Kesamaan kemiripan

Diff = Nilai dari jarak string asli ke string pembandingan

CS = String asli

ST = String pembandingan

Max(CS,ST) = Jarak string terbesar dari string pembandingan dan string asli

Algoritma Jaro Winkler Distance

Menurut Kurniawati, Sulistyono dan Sazali (2010) Algoritma Jaro Winkler Distance yaitu algoritma untuk mengukur kesamaan antara dua string, biasanya algoritma ini digunakan di dalam pendeteksian duplikat. Semakin tinggi Jaro-Winkler distance untuk dua string maka semakin mirip dengan string tersebut. Nilai normalnya ialah 0 menandakan tidak ada kesamaan dan 1 yang menandakan adanya kesamaan.

Dasar dari algoritma ini memiliki tiga bagian :

1. Menghitung panjang string.
2. Menemukan jumlah karakter yang sama di dalam dua string.
3. Menemukan jumlah transposisi

Pada algoritma Jaro digunakan rumus untuk menghitung jarak (d_j) antara dua string yaitu s_1 dan s_2 adalah

$$d_j = \frac{1}{3} + \left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m} \right) \quad (2)$$

Dimana :

m = jumlah karakter yang sama persis

s_1 = panjang string 1

s_2 = panjang string 2

t = jumlah transposisi

Jaro Winkler Distance menggunakan prefix scale (p) yang memberikan tingkat penilaian yang lebih dan prefix length (l) yang menyatakan panjang awalan yaitu panjang karakter yang

sama dengan string yang dibandingkan sampai ditemukannya ketidaksamaan. Bila string s_1 dan s_2 yang diperbandingkan, maka Jaro Winkler Distance-nya (d_w) adalah :

$$d_w = \left(d_j + \left(lp(l - d_j) \right) \right) \times 100\%$$

Dimana :

d_j = Jaro distance untuk string s_1 dan s_2

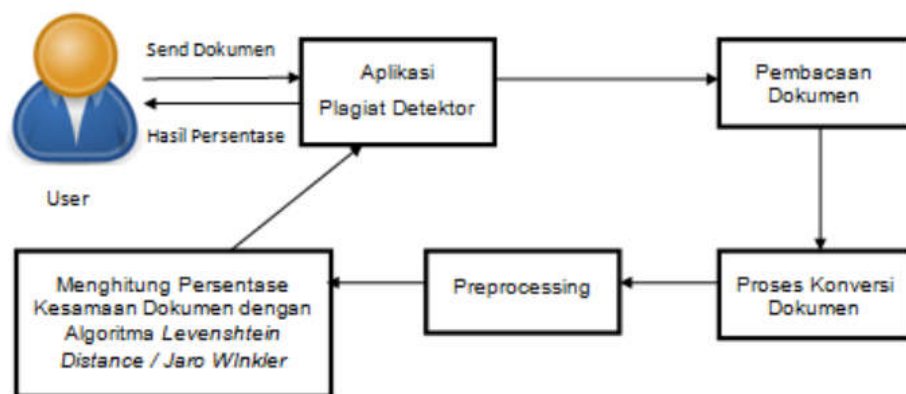
l = panjang prefiks umum diawal string (panjang karakter yang sama sebelum ditemukan ketidaksamaan max 4)

p = konstanta scaling factor .Nilai standar untuk konstanta ini menurut Winkler (p) = 0.1

ANALISIS DAN DESAIN SISTEM

Tindakan plagiarisme yang terjadi dalam dunia pendidikan tidak dapat dihindari, namun bisa dicegah dengan melakukan pendeteksian terhadap dokumen yang ingin dibandingkan. Pendeteksian yang dilakukan masih bersifat manual sehingga tidak efektif dan lambat dalam pengerjaannya. Oleh karena itu, dibutuhkan suatu sistem yang dapat membantu dalam melakukan pendeteksian terhadap dokumen teks secara efektif dan cepat. Dalam membuat sistem tersebut dibutuhkan suatu metode atau algoritma agar sistem dapat bekerja. Algoritma-algoritma yang dapat digunakan untuk melakukan pencocokan string yaitu algoritma *Levenshtein Distance* dan *Jaro Winkler*. Kedua algoritma tersebut dapat mengukur tingkat kemiripan dari kedua dokumen yang dibandingkan. Akan tetapi analisis perbandingan kinerja antara algoritma *Levenshtein Distance* dan *Jaro-Winkler* untuk deteksi plagiat dokumen belum pernah dilakukan. Karena itu, penulis ingin melakukan analisis perbandingan terhadap kedua algoritma tersebut sehingga dapat diketahui algoritma mana yang lebih akurat dan tepat dalam mendeteksi kesamaan dokumen.

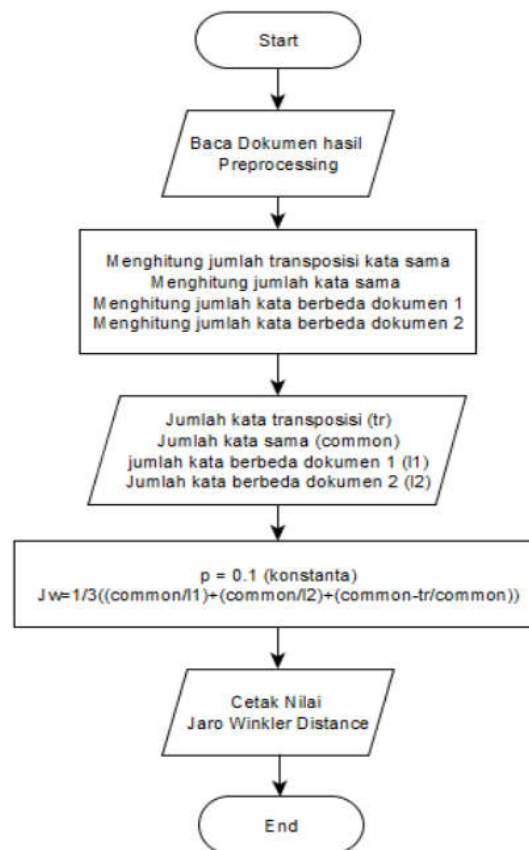
RANCANGAN SISTEM



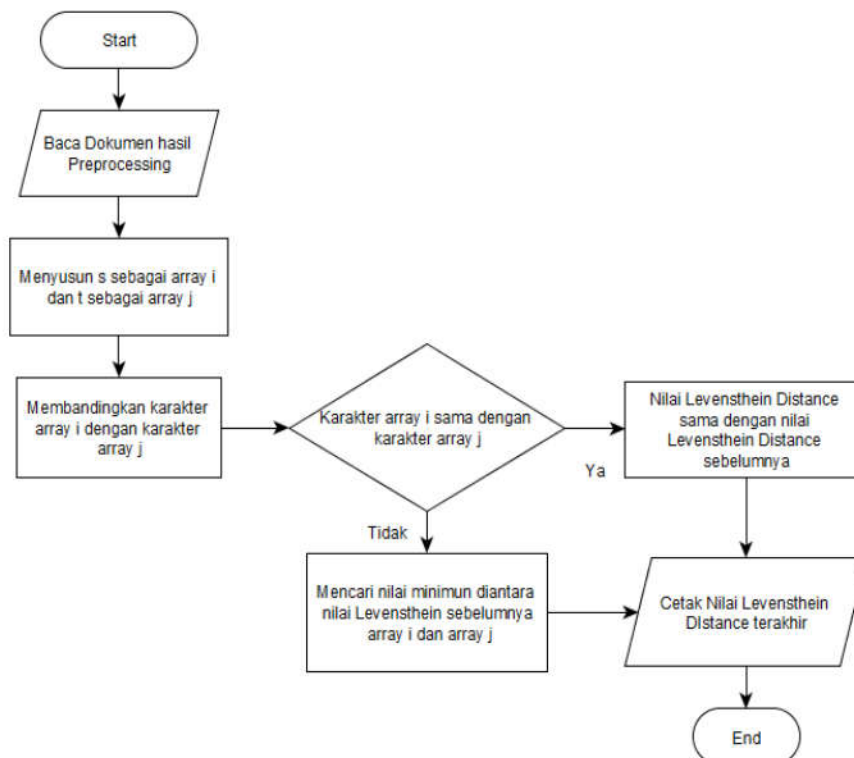
Gambar 1 Arsitektur Aplikasi

User menginput dokumen ke aplikasi plagiat detector, selanjutnya dokumen dibaca lalu di konvert menjadi format (.txt), setelah dikonvert dokumen tersebut dipreprocessing untuk memudahkan saat pembacaan algoritma. Setelah itu, hitung presentase kemiripannya

menggunakan algoritma *Levensthein Distance/Jaro Winkler*. Kemudian hasil persentase kemiripannya ditampilkan ke aplikasi *plagiat detector*.



Gambar 2 Flowchart Algoritma Jaro Winkler



Gambar 3 Flowchart Algoritma Levensthein Distance

RANCANGAN KEBUTUHAN SISTEM

Rancangan Kebutuhan sistem analisa perbandingan algoritma Jaro Winkler dan Levensthein Distance untuk deteksi plagiarisme dokumen sebagai berikut ;

1. Data uji (text data) similarity , data uji yang digunakan sebanyak 5 buah dokumen. Data yang digunakan sebagai bahan pengujian diambil dari jurnal yang ditulis oleh Thalib dan Kusumawati (2010). Dalam pengujiannya, dilakukan perbandingan terhadap lima dokumen teks menggunakan aplikasi deteksi plagiarisme. Dimana kelima dokumen tersebut mempunyai panjang dan isi yang berbeda. Dokumen ini diberi nama A1, A2, A3, A4 dan A5. Dokumen A1 berisi informasi atau kata-kata yang sama dengan A2, hanya saja dilakukan perubahan struktur kalimat dalam paragrafnya. Dokumen A3 memiliki isi yang berbeda dengan dokumen A1 dan A2 tetapi memiliki tema yang sama. Pada dokumen A3 dilakukan perubahan struktur kata di dalam kalimat dengan dokumen A4. Sedangkan dokumen A5 merupakan gabungan dari dokumen A1 dan dokumen A4. Untuk mempermudah proses identifikasi, data gambar dan tabel tidak diikutsertakan dalam proses perbandingan. Berikut Tabel Data dokumen uji.

Tabel 1 Data Dokumen Uji Similarity

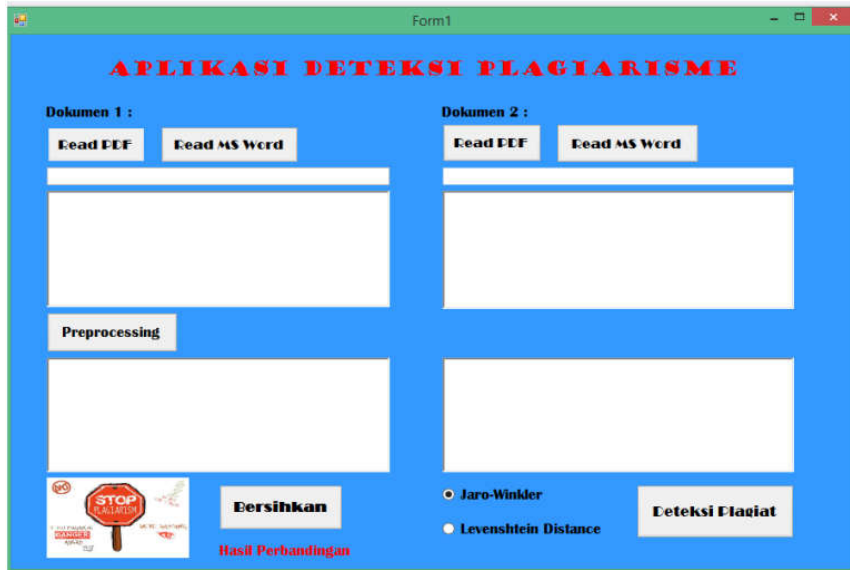
No	Dokumen	Kalimat
1.	A1	Akustik adalah ilmu yang mempelajari perilaku bunyi dan sangat penting pada ruangan. Dinding yang keras dan polos dari sebuah ruangan akan memantulkan bunyi dan membuat ruangan tersebut bergema. Ruangan yang kecil akan terbantu mencegah hal ini bila ada bahan pada dinding dan langit-langit yang menyerap bunyi. Tirai dan karpet yang tebal juga akan membantu. Pada ruangan yang besar seperti gedung konser, diperlukan permukaan yang halus dan keras di belakang para pemain atau penyanyi untuk membantu membawa bunyi ke arah penonton, dan bahan yang menyerap bunyi di belakang gedung untuk mencegah gema.
2.	A2	Ruangan yang kecil akan terbantu mencegah hal ini bila ada bahan pada dinding dan langit-langit yang menyerap bunyi. Pada ruangan yang besar seperti gedung konser, diperlukan permukaan yang halus dan keras di belakang para pemain atau penyanyi untuk membantu membawa bunyi ke arah penonton, dan bahan yang menyerap bunyi di belakang gedung untuk mencegah gema. Akustik adalah ilmu yang mempelajari perilaku bunyi dan sangat penting pada ruangan. Dinding yang keras dan polos dari sebuah ruangan akan memantulkan bunyi dan membuat ruangan tersebut bergema. Tirai dan karpet yang tebal juga akan membantu.
3.	A3	Sebagaimana sebuah daerah pada umumnya, Lampung memiliki beraneka ragam jenis musik, mulai dari jenis tradisional hingga modern, yang mengadopsi kebudayaan musik global. Adapun jenis musik yang masih bertahan hingga sekarang adalah Klasik Lampung. Jenis musik ini biasanya diiringi oleh alat musik gambus dan gitar akustik. Jenis musik ini merupakan perpaduan budaya Islam dan budaya asli itu sendiri. Beberapa kegiatan festival diadakan untuk mengembangkan budaya musik tradisional tanpa harus khawatir akan kehilangan jati diri. Festival Krakatau contohnya, adalah sebuah Festival yang diadakan oleh Pemda Lampung yang bertujuan untuk mengenalkan Lampung kepada dunia luar dan sekaligus menjadi ajang promosi pariwisata.
4.	A4	Beragam jenis musik dimiliki oleh daerah Lampung, sebagaimana sebuah daerah pada umumnya, mulai dari jenis tradisional hingga modern, yang mengadopsi kebudayaan musik global. Klasik Lampung adalah jenis musik yang masih bertahan hingga sekarang. Alat musik Gambus dan gitar akustik biasanya mengiringi jenis musik ini. Jenis musik ini merupakan perpaduan budaya Islam dan budaya asli itu sendiri. Budaya musik tradisional dikembangkan tanpa harus khawatir akan kehilangan jati diri dengan cara mengadakan beberapa kegiatan festival. Pemda Lampung mengadakan sebuah festival yang bertujuan untuk mengenalkan Lampung kepada dunia luar dan sekaligus menjadi ajang promosi pariwisata, dinamakan Festival Krakatau.
5.	A5	Beragam jenis musik dimiliki oleh daerah Lampung, sebagaimana sebuah daerah pada umumnya, mulai dari jenis tradisional hingga modern, yang mengadopsi kebudayaan musik global. Klasik Lampung adalah jenis musik yang masih bertahan hingga sekarang. Alat musik Gambus dan gitar akustik biasanya mengiringi jenis musik ini. Jenis musik ini merupakan perpaduan budaya Islam dan budaya asli itu sendiri. Budaya musik tradisional dikembangkan tanpa harus khawatir akan kehilangan jati diri dengan cara mengadakan beberapa kegiatan festival. Pemda Lampung mengadakan sebuah festival yang bertujuan untuk mengenalkan Lampung kepada dunia luar dan sekaligus menjadi ajang promosi pariwisata, dinamakan Festival Krakatau. Akustik adalah ilmu yang mempelajari perilaku bunyi dan sangat penting pada ruangan. Dinding yang keras dan polos dari sebuah ruangan akan memantulkan bunyi dan membuat ruangan tersebut bergema. Ruangan yang kecil akan terbantu mencegah hal ini bila ada bahan pada dinding dan langit-langit yang menyerap bunyi. Tirai dan karpet yang tebal juga akan membantu. Pada ruangan yang besar seperti gedung konser, diperlukan permukaan yang halus dan keras di belakang para pemain atau penyanyi untuk membantu membawa bunyi ke arah penonton, dan bahan yang menyerap bunyi di belakang

2. Data uji waktu proses, data uji yang digunakan untuk pengujian waktu proses terdiri dari 4 file dokumen. Masing-masing file dokumen memiliki ukuran file dan banyak kata yang berbeda.

Tabel 2 Dokumen Data Uji Waktu Proses

No	Dokumen	Ukuran (kb)	Kata
1	B1	13	255
2	B2	13	303
3	B3	13	431
4	B4	14	561

3. Aplikasi deteksi plagiarisme dokumen untuk menguji nilai similarity dan waktu proses dari algoritma Jaro Winkler dan Levenstien Distance. Berikut contoh tampilan aplikasi yang digunakan dalam pengujian;



Gambar 4 Tampilan Aplikasi Deteksi Plagiarisme

PENGUJIAN SISTEM

Pada tahap ini akan dilakukan pengujian terhadap algoritma Jaro Winkler dan Levenstien Distance yang telah diimplementasikan ke dalam sistem. Kemudian kedua algoritma tersebut diuji lalu dilakukan analisis perbandingan terhadap kedua algoritma tersebut berdasarkan keakuratan dan waktu proses yang dibutuhkan. Proses pengujian ini dilakukan pada sebuah komputer yang tidak terhubung pada jaringan dengan menggunakan aplikasi deteksi plagiarisme yang berbasis desktop untuk membandingkan kedua algoritma tersebut.

Dengan memanfaatkan perbandingan hasil keluaran algoritma Jaro Winkler dengan algoritma Levenstien Distance yang telah dibantu dengan proses preprocessing, maka dapat dilihat perbedaan akurasi dan performanya. Berdasarkan data pengujian, maka akan dilakukan pengujian terhadap similarity dan waktu proses yang dibutuhkan dalam proses perbandingan.

1. Pengujian similarity

Pengujian ini ditujukan untuk melihat similarity dari algoritma Jaro Winkler maupun algoritma Levenshtein Distance dalam membandingkan dua dokumen. Dokumen yang digunakan untuk pengujian ini terdiri dari lima dokumen yang mempunyai panjang dan konteks yang berbeda. Pada pengujian ini dilakukan perbandingan terhadap dua dokumen yang memiliki isi sama, dokumen yang memiliki isi berbeda, dokumen yang memiliki isi sama

namun mengalami perubahan struktur kata dan struktur kalimat, dan dokumen yang mengalami penggabungan.

2. Pengujian waktu proses

Selain pengujian similarity, juga dilakukan pengujian waktu proses. Pengujian ini ditujukan untuk mengamati secara eksplisit waktu proses yang dibutuhkan oleh kedua algoritma yang diuji. Data yang digunakan didalam pengujian ini terdiri dari 4 dokumen yang mempunyai jumlah kata dan ukuran berbeda.

HASIL PENGUJIAN

Pada contoh implementasi enkripsi dan dekripsi file dengan algoritma blowfish akan dilakukan pada beberapa tipe file yakni file dengan ekstensi .txt, .pdf, .jpg, dan .docx. Penulis membandingkan waktu enkripsi menggunakan fungsi enkripsi PHP yaitu mcrypt dengan kode enkripsi yang ditulis sendiri oleh penulis.

Dari pengujian yang telah dilakukan maka diperoleh hasil pengujian sebagai berikut :

1. Hasil Pengujian Similarity

Memperlihatkan hasil pengujian persentase kemiripan (similarity) dokumen dari dua kali percobaan yang terdiri dari percobaan terhadap Algoritma Jaro Winkler dan percobaan terhadap Algoritma Levensthein Distance.

Tabel 3 Hasil Pengujian Similarity Data Algoritma Jaro WInkler

Dokumen	A1 (%)	A2(%)	A3(%)	A4(%)	A5(%)
A1	100	84.59	74.63	74.92	66.57
A2	84.59	100	74.63	74.92	66.6
A3	74.63	74.63	100	83.73	70
A4	74.92	74.92	83.73	100	90.91
A5	66.57	66.6	70	90.91	100

Tabel 4 Hasil Pengujian Similarity Data Algoritma Levensthein Distance

Dokumen	A1 (%)	A2(%)	A3(%)	A4(%)	A5(%)
A1	100	28.71	27.64	27.77	45.18
A2	28.71	100	28.13	28.26	33.27
A3	27.64	28.13	100	58.54	35.98
A4	27.77	28.26	58.54	100	54.44
A5	45.18	33.27	35.98	54.55	100

Dari pengujian terhadap kedua algoritma tersebut terlihat bahwa struktur kalimat dan struktur kata yang berbeda mempengaruhi nilai similarity suatu dokumen.

2. Hasil Pengujian Waktu Proses

Memperlihatkan perbandingan waktu proses dari percobaan terhadap Algoritma *Jaro Winkler* dan percobaan terhadap Algoritma *Levensthein Distance*.

Tabel 5 Hasil Pengujian Waktu Proses Algoritma Jaro Winkler

Dokumen	B1 (detik)	B2 (detik)	B3 (detik)	B4 (detik)
B1	0.031	0.031	0.031	0.031
B2	0.031	0.046	0.046	0.062
B3	0.031	0.046	0.079	0.091
B4	0.036	0.050	0.092	0.131

Tabel 6 Hasil Pengujian Waktu Proses Algoritma Levensthein Distance

Dokumen	B1 (detik)	B2 (detik)	B3 (detik)	B4 (detik)
B1	0.062	0.078	0.109	0.146
B2	0.078	0.093	0.109	0.156
B3	0.093	0.129	0.190	0.245

B4	0.133	0.172	0.246	0.311
----	-------	-------	-------	-------

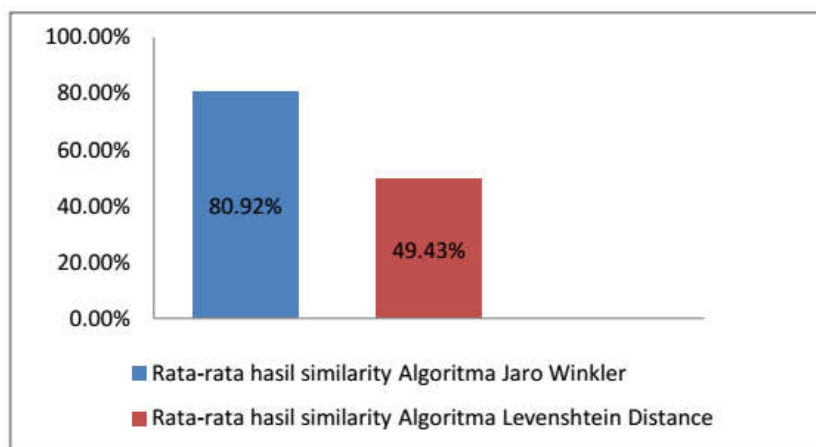
Dapat dilihat bahwa nilai similarity dan waktu proses yang dihasilkan kedua algoritma tersebut terlihat jelas perbedaannya.

ANALISIS PENGUJIAN

Berdasarkan hasil data pengujian yang telah dikumpulkan terhadap algoritma Jaro Winkler dan Levenshten Distance, maka dilakukan analisis perbandingan antara kedua algoritma tersebut.

Untuk analisis perbandingan similarity-nya, menggunakan data hasil pengujian similarity Jaro Winkler dari tabel 3 dan data hasil pengujian Levensthein Distance dari tabel 4 Kemudian dari kedua data tersebut masing-masing dihitung rata-rata similarity yang dihasilkan.

Dimana untuk algoritma Jaro Winkler nilai rata-rata similarity yang dihasilkan yaitu sebesar 80.92 %, sedangkan untuk algoritma Levensthein Distence rata-rata nilai similarity- nya yaitu sebesar 49.43%.

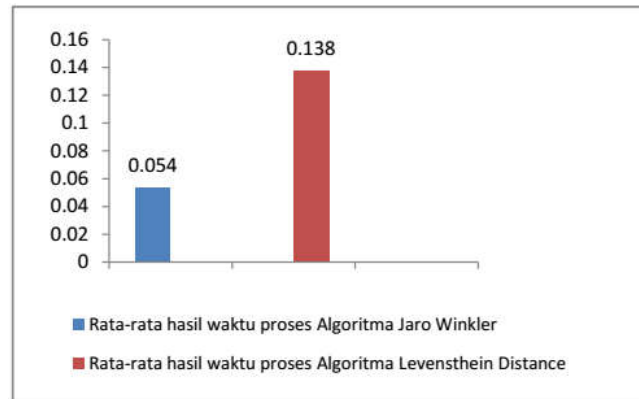


Gambar 5 Grafik Rata-rata Hasil Perbandingan Similarity Algoritma

Berdasarkan analisis perbandingan yang telah dilakukan, pada Gambar 5 terlihat jelas perbedaan persentase similarity yang cukup significant antara algoritma Jaro Winkler dan Levensthein Distance.

Algoritma Jaro Winkler memiliki rata-rata similarity lebih tinggi dibandingkan dengan Levenshtien Distance yaitu sebesar 80.92%, sedangkan rata-rata similarity Levensthein Distance yaitu hanya 49.43%. Ini membuktikan bahwa algoritma Jaro Winkler lebih akurat dalam mendeteksi kesamaan dokumen dibanding dengan Algoritma Levensthein Distance.

Sedangkan untuk data hasil pengujian waktu proses, dihitung nilai rata-ratanya dari tabel 5 dan tabel 6. Dimana untuk algoritma Jaro Winkler rata-rata waktu prosesnya yaitu sebesar 0.054 detik, sedangkan untuk algoritma Levensthein Distance yaitu sebesar 0.138 detik.



Gambar 6 Grafik Rata-rata Hasil Perbandingan Waktu Proses Algoritma

Berdasarkan hasil pengujian tersebut, dapat dilihat pada Gambar 6 bahwa Algoritma Jaro Winkler memiliki waktu proses yang lebih cepat dibanding dengan algoritma Levensthein Distance dalam mendeteksi plagiarisme kedua dokumen.

KESIMPULAN

Berdasarkan hasil penelitian yang diperoleh, maka penulis dapat menarik beberapa kesimpulan berikut, antara lain:

1. Perbandingan kinerja algoritma Jaro Winkler Distance dan algoritma Levensthein Distance diukur dengan menggunakan 2 data uji yaitu data uji untuk mengukur nilai similarity dan data uji untuk mengukur waktu proses. Selanjutnya data uji tersebut diujikan menggunakan aplikasi deteksi plagiarisme untuk mengetahui berapa nilai similarity dan waktu proses yang dihasilkan dari pengujian tersebut. Selanjutnya dari hasil pengujian tersebut dianalisis perbandingannya, dan analisis perbandingan yang didapat dari pengujian nilai similarity, yaitu algoritma Jaro Winkler memiliki nilai similarity yang lebih akurat yaitu sebesar 80.92% dibanding dengan algoritma Levensthein Distance yaitu hanya sebesar 49.43%. Sedangkan untuk pengujian waktu proses, algoritma Jaro Winkler memiliki rata-rata waktu proses yang lebih cepat yaitu 0.054 detik, dibandingkan algoritma Levensthein Distance yaitu 0.138 detik.
2. Peneliti menemukan bahwa algoritma Jaro Winkler mampu membandingkan dokumen yang mengalami perubahan struktur kalimat atau struktur kata, sedangkan algoritma Levensthein Distance hanya akan bekerja lebih baik jika kedua dokumen yang dibandingkan mempunyai urutan atau struktur kata/kalimat yang sama.
3. Dalam mendeteksi plagiarisme dokumen, peneliti lebih menyarankan algoritma Jaro Winkler Distance untuk dipakai dalam pendeteksian plagiarisme jika dibandingkan dengan algoritma Levensthein Distance.

DAFTAR PUSTAKA

- [1] Gottschalk, L. 1986. Understanding History; A Primer of Historical Method (terjemahan Nugroho Notosusanto). Jakarta : UI Press.
- [2] Hasanudin, M. S. 2014. Pengenalan Bachmann-Landau Big O Notation dan Penggunaannya.
- [3] Program Studi Teknik Informatika, Universitas Komputer Indonesia. Bandung Heribertius. 2012. Analisa Algoritma 1.

-
- [4] <http://heri.staff.unisbank.ac.id/2012/02/21/analisa-algoritma-1-semester-pendek-ganjil-2012/>. Diakses pada 19 November 2016
- [5] Junedy, R. 2014. Perancangan Aplikasi Deteksi Kemiripan Isi Dokumen Teks dengan Menggunakan Metode Levenshtein Distance. Jurusan Teknik Informatika STMIK Budi Darma. Medan.
- [6] KBBI. 2013. Plagiarisme. <http://kbbi.web.id/plagiarisme>. Diakses pada 15 Mei 2016.
- [7] Kurniawati, A., Puspi, S., & Rahman, S. 2010. Implementasi Algoritma Jaro-Winkler Distance untuk Membandingkan Kesamaan Dokumen Berbahasa Indonesia. Universitas Gunadarma.
- [8] Soelistyo, H. 2011. Plagiarisme: Pelanggaran Hak Cipta dan Etika. Yogyakarta : Penerbit Kanisius.
- [9] Sulianta, F. 2007. Seri referensi praktis: Konten internet, 71-72. Diunduh dari <http://books.google.com/books?isbn=9792717889>. Diakses pada 16 Mei 2016.
- [10] Thalib, F., & Ratih K. 2010. Pembuatan Program Aplikasi untuk Pendeteksian Kemiripan Dokumen Teks dengan Algoritma Smith-Waterman. Universitas Gunadarma. Depok.
- [11] Triawati, C. 2009. Metode Pembobotan Statistical Concept Based untuk Klastering dan Kategorisasi Dokumen Berbahasa Indonesia. IT TELKOM Bandung.