Deteksi Plagiarisme Artikel Jurnal menggunakan *Latent Semantic Analysis* (LSA)

Kamal Fauzan Navaro[1], Septiyawan Rosetya Wardhana[2], Rinci Kembang Hapsari[3]
Institut Teknologi Adhi Tama Surabaya[1,2,3]
e-mail corresponding author: rincikembang@itats.ac.id

ABSTRACT

Nowadays, plagiarism is no longer strange for university students, especially students. The problem faced is minimizing the occurrence of plagiarism among students, which continues to increase daily. Apart from that, plagiarism also violates academic ethics and can reduce student competence. Apart from that, plagiarism is included in stealing other people's written work. Therefore, plagiarism must be stopped immediately. One solution to this problem is to create an application that can detect document plagiarism effectively and efficiently. This application is expected to reduce the occurrence of plagiarism by detecting similarities in documents, one of which is a journal. This plagiarism detection application was built using the Latent Semantic Analysis (LSA) method. Based on the description above, a Document Similarity Analysis for Journal Plagiarism Detection Using Web-Based LSA was created. System testing was carried out to measure the performance of the Latent Semantic Analysis (LSA) method used in the system on 200 training data and 20 testing data, resulting in an average accuracy percentage of 87.88%. The percentage of accuracy obtained is quite large, so the system created is quite good.

Kata kunci: *Plagiarism, Journals, Web, Latent Semantic Analysis (LSA), Accuracy Testing*

ABSTRAK

Plagiarime saat ini bukan lagi suatu hal yang asing bagi para golongan pelajar perguruan tinggi terutama mahasiswa. Permasalahannya adalah meminimalisir plagiarisme dikalangan mahasiswa yang semakin hari semakin meningkat. Selain itu, plagiarisme juga merupakan pelanggaran etika akademik dan dapat menurunkan keterampilan mahasiswa. Selain itu, plagiarisme termasuk dalam kategori pencurian karya tulis orang lain. Karena itu, plagiarisme harus segera dihentikan. Salah satu solusi dari permasalahan tersebut adalah membuat sebuah aplikasi yang dapat melakukan deteksi plagiarisme dokumen secara efektif dan efisien. Aplikasi ini diharapkan dapat mengurangi terjadinya plagiarisme melalui deteksi persamaan dokumen salah satunya adalah jurnal. Aplikasi deteksi plagiarisme ini dibangun dengan mengimplementasikan metode Latent Semantic Analysis (LSA). Berdasarkan uraian diatas maka dibuatlah Analisa Kesamaan Dokumen Untuk Deteksi Plagiarisme Jurnal Menggunakan LSA Berbasis Web. Pengujian sistem dilakukan untuk mengukur kinerja dari metode Latent Semantic Analysis (LSA) yang digunakan pada sistem pada 200 data training dan 20 data testing maka didapatkan rata – rata persetase akurasi sebesar 87.88%. Persetase akurasi yang didapatkan cukup besar sehingga dapat dikatakan bahwa sistem yang dibuat cukup bagus.

Kata kunci: Plagiarisme, Jurnal, Web, *Latent Semantic Analysis* (LSA), Pengujian Akurasi

PENDAHULUAN

Plagiarisme adalah tindakan mengambil karya dan gagasan orang lain tanpa menyebutkan sumbernya dan mengakuinya sebagai karya sendiri. Dikalangan mahasiswa masih sering terjadi tindak plagiarism. Berdasarkan "Peraturan Menteri Pendidikan Nasional Republik Indonesia Nomor 17 Tahun 2010 Tentang Pencegahan dan Penanggulangan Plagiat Di Perguruan Tinggi", seringnya kejadian tersebut perlu adanya pengecekan dokumen karya mahasiswa untuk menghindari hal tersebut (Risparyanto, 2020).

Plagiarisme adalah tindak pelanggaran etika akademik yang dapat menurunkan kompetensi mahasiswa. Selain itu, plagiarisme termasuk dalam kategori pencurian karya tulis orang lain. Karena itu, plagiarisme harus segera dihentikan. Dalam upaya mengurangi plagiarisme perlu dikembangkan sebuah sistem otomatisasi deteksi plagiarisme.

Banyak penelitian yang telah dilakukan mengenai plagiarisme, antara lain oleh Nurdin dkk yang membangun sistem pendeteksi kemiripan antara dokumen teks yang memiliki perbedaan. Dimana dokumen yang digunakan tersebut menggunakan bahasa Indonesia, dan file dokumen berformat doc, rtf, docx, dan pdf. Pada proses klasifikasi kemiripan menggunakan metode w*eight tree,* dengan tingkat keakuratan mencapai 90%[1]. Imam Nawawi dkk telah membuat aplikasi pendeteksi plagiarisme pada dokumen tugas akhir dengan nama Doristec dengan menggunakan metode LCS (*Longest Common Subsequence*). Sistem dimodifikasi untuk mencapai hasil yang konsisten dengan perancangan. LCS dapat melakukan membandingkan lebih adri dua dokumen pembanding, dengan menguji lebih dari satu kalimat calon pembanding. Dimana hasil pengujian sistem deteksi plagiarisme mendapatkan nilai akurasi yang tinggi[2].

Pratama dkk mengembangkan deteksi plagiarisme artikel jurnal, dimana tahapan proses sistem dengan melakukan perbandingan artikel jurnal yang sudah diunggah pada repositori hasil *grabbing data* DOAJ. Dengan menggunakan metode *cosine similarity* dilakukan untuk menghitung nilai presentase kesamaan antar artikel. Selain itu juga dihitung nilai kesamaan artikel jurnal antar publisher yang ada di *repository*. Dari hasil pengujian didapatkan nilai *recall* 13% sedangan nilai precisionnya 8%[3].

Pada literatur yang lain, pencarian berbasis metadata memiliki kelemahan yaitu tidak dapat menemukan dokumen yang serupa. Dengan pencarian semantik, kelemahan ini dapat diatasi, seperti yang diterapkan oleh Azharyani dkk. Pencarian semantik dengan memahami maksud pencari dan makna kontekstual dari istilah tersebut. Pencarian dilakukan dengan menggunakan kombinasi LSA (*Latent Semantic Analysis*) dengan WTS (*Weighted Tree Similarity*). Pencarian tersebut menghasilkan nilai presisi rata-rata sebesar 57,12% dan rata-rata *recall rate* sebesar 85,08% yang menunjukkan tingkat keberhasilan sistem dalam menemukan dokumen yang relevan.[4].

Metode LSA dapat digunakan untuk menyusun sebuah ringkasan berdasarkan dari data artikel yang dikumpulkan dengan metode *scrapping*. Dimana LSA dapat membantu mendapatkan makna tersembunyi dari kumpulan kalimat. Dimana penyusunan ringkasan digabungkan dengan metode *cross*. Hasil penelitian menunjukan bahwasannya metode LSA yang digabungkan dengan metode *cross* bisa digunakan dalam penyusunan ringkasan otomatis secara baik. Pengumpulan dataset dilakukan pada bulan Februari sampai dengan Juni 2020, yang diambil 120 dokumen. Dataset dibagi dua dengan 90% sebagai data pembelajaran dan 10% sebagai data pengujian. Pengujian yang dilakukan dengan nilai compression rate sebesar 30% . Pengujian menghasilkan nilai f-measure sebesar 90.68% dan nilai rata-rata *recall* sebesar 85% [5].

Berdasarkan penelitian terdahulu, sehingga dalam upaya mengurangi tingkat plagiarisme ditingkat mahasiswa, dalam penelitian ini mengembangkan deteksi plagiarism artikel jurnal dengan menggunakan metose LSA.

TINJAUAN PUSTAKA

Plagiarisme

Plagiarisme berasal dari kata plagiarius, merupakan Bahasa latin yang memiliki arti mencuri atau mengambil. Plagiarisme adalah perilaku pencurian intelektual atau kebohongan. Terdapat pengertian plagiarisme berdasarkan pendapat beberapa ahli, antara lain: Plagiarisme adalah publikasi yang dilakukan oleh seorang ilmuwan atau seniman atas suatu karya ilmu pengetahuan atau seni kepada masyarakat umum atas seluruh atau sebagian besar karya dengan tidak menyebut nama pengarangnya.

Plagiarisme adalah suatu tindakan menyalahgunakan, mencuri atau menyita, menerbitkan, berbicara, mengklaim sebagai miliknya sendiri suatu gagasan, karya atau ciptaan yang aslinya milik orang lain. Secara umum plagiarisme dapat diartikan sebagai mengambil gagsan (pendapat) orang lain serta menjadikannya seolah-olah merupakan tulisan atau pendapatnya

sendiri, misalnya dengan memposting karya orang lain atas namanya sendiri. Orang yang melakukan plagiarisme disebut plagiator atau penjiplak. Sanksi dan hukuman bagi pelaku plagiarisme diatur dalam Pasal 25 Ayat 2 dan Pasal 70 Undang-Undang Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional dan Penjelasannya. Dengan demikian jelas bahwa plagiarisme adalah suatu perbuatan menyimpang yang melanggar hukum dan tidak dapat ditoleransi karena mencuri karya atau hak cipta orang lain. Terdapat beberapa tipe plagiarisme, yaitu[6]:

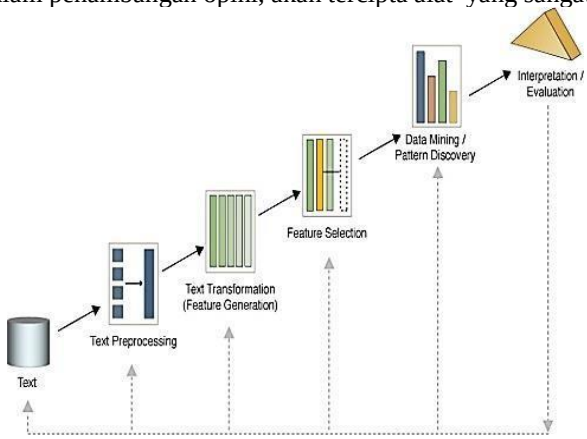
a. Plagiarisme langsung (direct plagiarism). Plagiarisme langsung jika penulis langsung menyalin bagian atau seluruh artikel dan bagian tesebut disajikan tanpa memberikan informasi kutipan atau informasi dari karya orang lain.
b. Plagiarisme implisit (*misquotation*), plagiarisme jenis ini terjadi ketika seorang penulis mengutip sebagian dari suatu karya tulis tetapi tidak menentukan di mana kutipan itu dimulai dan diakhiri.
c. Plagiarisme mosaik, yaitu pengarang mengutip sebagian suatu karya tulis dengan memodifikasi kata-katanya sendiri, meskipun hanya memodifikasi kata-kata tertentu..

Jenis-jenis plagiarisme terbagi dalam banyak golongan, yaitu:

1. Jenis-jenis plagiarisme berdasarkan pada aspek pencuriannya antara lain plagiarisme terhadap ide, isi (dataset penelitian), kata, kalimat, paragraf, dan plagiarisme keseluruhan.
2. Jenis plagiarisme bergantung pada apakah plagiarisme tersebut disengaja atau tidak.
3. Jenis plagiarisme didasarkan pada rate atau persentase kata, kalimat, dan paragraf bajakan, plagiarisme tergolong ringan jika mencapai 70%.
4. Jenis plagiarisme didasarkan pada model plagiarisme, khususnya plagiarisme verbatim dan plagiarisme mosaik.

Text Mining

Text mining tidak bisa dipisahkan, terutama ketika menganalisis sentimen di media online. Teks mining adalah studi data mining yang lebih spesifik yang bertujuan untuk menemukan pola tersembunyi dalam teks, oleh karena itu, ketika menggabungkan analisis sentimen dan penambangan teks dalam penambangan opini, akan tercipta alat yang sangat efisien dan andal[7].



Gambar 1. Proses Text Mining

Pada Gambar 1 merupakan tahap prosestext mining secara umum meliputi langkah-langkah pengumpulan teks, inisialisasi atau preprosesing teks, transformasi teks, pemilihan fitur atau ekstraksi, data mining dan interpretasi data. Semua langkah ini akan menjadi referensi dalam penelitian yang dilakukan. Berikut merupakan tahap *preprocessing* dari *text mining*[8]:

1. Tokenizing, Tahap *tokenizing* merupakan proses pemotongan kalimat-kalimat dalam dokumen menjadi kata per kata. Tahap ini dilakukan untuk menghilangkan karakter atau simbol tertentu seperti tanda baca, angka, serta memfilter berdasarkan panjang teks.
2. *Case Folding*, berfungsi untuk menubah huruf besar menjadi huruf kecil. Sehingga, seluruh kata dalam dokumen akan diproses dan diubah menjadi bentuk standar.
3. *Filtering, Filtering* adalah proses pemilihan atau pengambilan kata dan dilakukan dengan metode stopword atau wordlist. Dengan fitur ini, sebelum diklasifikasikan, teks yang tidak relevan dengan analisis akan dihilangkan. Hal ini dilakukan untuk memperkecil ukuran teks tanpa mengurangi isi teks.
4. *Stemming* adalah proses pencarian kata dasar dari hasil proses filtering menggunakan aturan tertentu. Dalam teks bahasa Indonesia bisa digunakan librari sastrawi.

***Latent Semantic Analysis* (LSA)**

Analisis semantik laten *(LSA)* adalah metode yang menggunakan model statistik matematis untuk menganalisis struktur semantik sebuah teks. LSA dapat digunakan untuk memberikan nilai dalam teks dengan cara mengubah teks tersebut menjadi sebuah matriks yang memberikan nilai pada setiap istilah guna mencari persamaan dengan istilah acuannya. Ada korpus di LSA, korpus adalah kumpulan dokumen dengan subyek yang sasma. Metode LSA menerima inputan berupa file teks, yang kemudian direpresentasikan sebagai sebuah matriks [9].

Pada tahun 1998 LSA diusulkan serta dipatenkan oleh Scott Deerwester, Thomas Launder dan Susan Dumais. Metode ini dimanfaatkan dalam bidang pengambilan informasi dan pemrosesan bahasa alami. Contoh penggunaan metode LSA adalah mencari nilai-nilai serupa dalam karya ilmiah. Pada metode LSA menerapkan teknik matematika aljabar linier, khususnya SVD (*Singular Value Decomposition).*

Dalam komputasi SVD, diperlukan term matriks yang sudah terbentuk sebelumnya. Perhitungan SVD bertujuan untuk mendapatkan tren atau keterhubungan baru antar kata. Hasil perhitungan SVD membentuk tiga buah matriks, yaitu dua buah matriks ortogonal dan sebuah matriks diagonal. Perhitungan SVD berdasarkan pemfaktoran suatu matriks A berukuran txd ditunjukkan pada persamaan (1).

$$A_{txd} = U_{txn} \times S_{nxn} \times V_{dxn^T} \tag{1}$$

Cosine Similarity

Cosinus Similarity digunakan dalam mencari nilai sudut kosinus diantara vektor dokumen dan vektor *query*. Semakin kecil nilai sudut yang diperoleh maka semakin tinggi kemiripan dokumennya. Dimana vektor dokumen mewakili dokumen/artikel yang diunduh, sedangkan vektor queri mewakili dokumen/artikel yang akan dibandingkan. Rumus *cosine-similarity* ditunjukkan seperti pada persamaan (2).[9].

$$\mathrm{s}\left(d_j, q_k\right) = \frac{\sum_{j=1}^{t} td_{ij} \times tq_{ij}}{\sqrt{\sum_{j=1}^{t}\left(td_{ij}\right)^2 \times \sum_{j=1}^{t}\left(tq_{ij}\right)^2}} \tag{2}$$

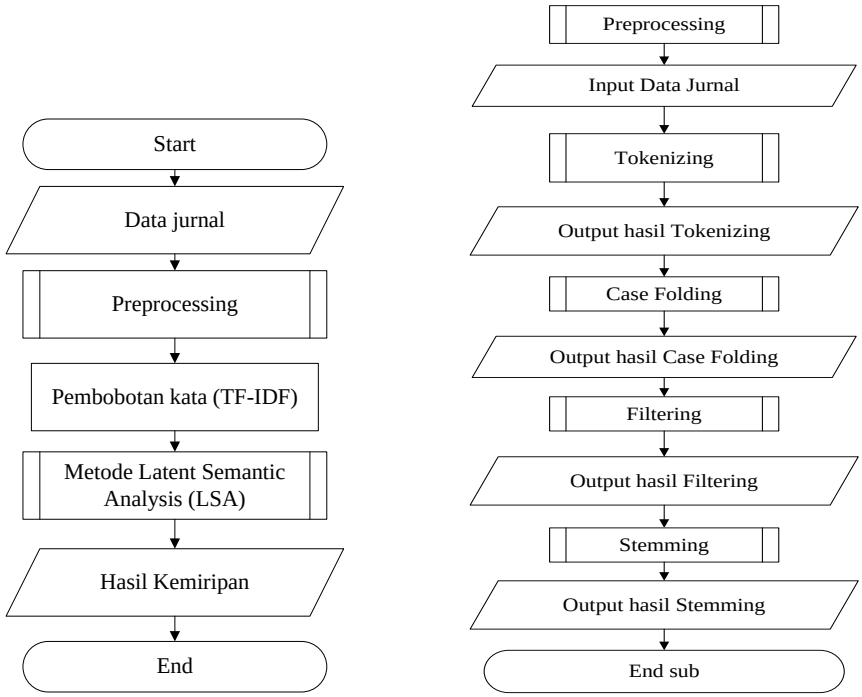
Semakin besar nilai hasil perhitungan Consine Similarity maka akan semakin dekat kesamaan antar artikel/dokumen tersebut. Nilai hasil perhitungan rumus *Consine Similarity* berada pada interval 0–1.

Dataset

Dataset yang digunkan dalam penelitian ini berisi dokumen yang berektensi *.pdf. Dataset diambil dari artikel jurnal skripsi mahasiswa. Dimana 500 artikel jurnal dijadikan sebagai data training.

METODE

Dalam penelitian ini perancangan alur sistem deteksi plagiarisme artikel jurnal menggunakan Latent Semantic Analysis (LSA) ditunjukkan oleh Gambar 2. Dimana Gambar 2 adalah flowchart sistem yang menjelaskan alur proses sistem Deteksi Kesamaan Jurnal. Pertama, melakukan proses input data jurnal yang akan dilatih. Kedua, melakukan proses *preprocessing* data teks.

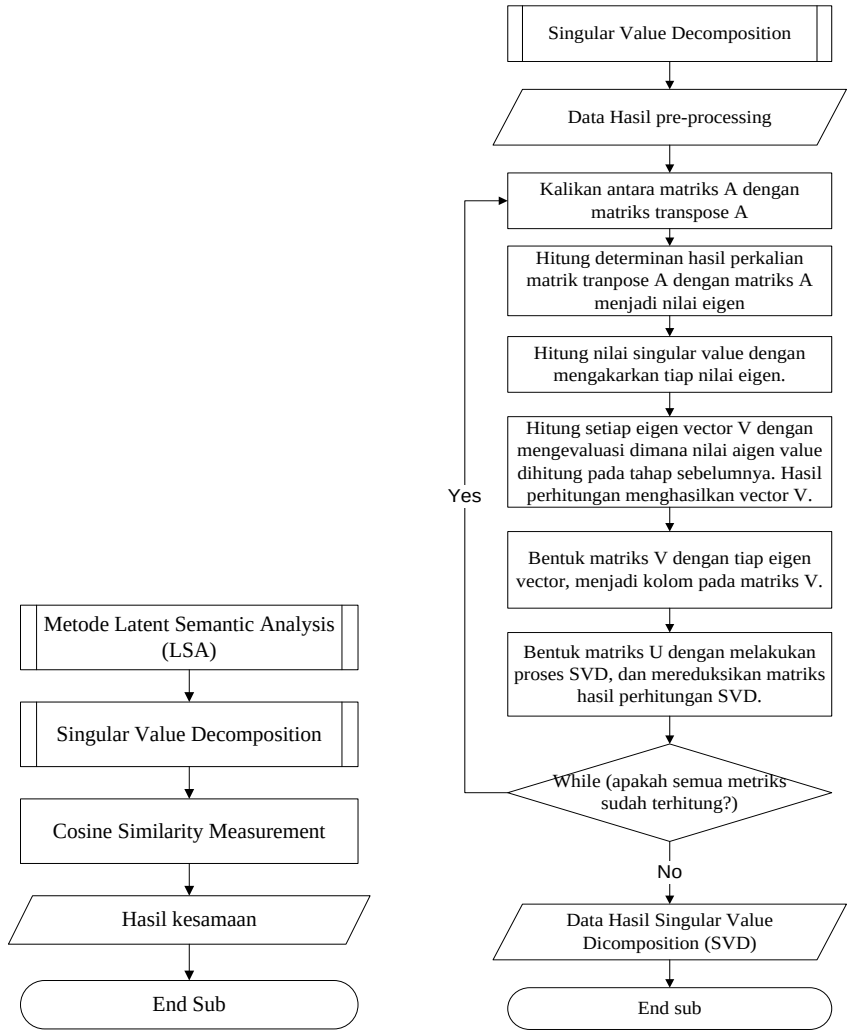


Gambar 2. (a) Alur Utama Sistem Deteksi Kesamaan Artikel jurnal (b) Alur proses *prepocessing*

Dimana preprocessing ini terdiri dari empat langkah yaitu, pertama proses *Tokenizing*, kedua proses *Case Folding*, ketiga proses *Filtering*, dan keempat proses *Stemming*. Proses *tokenizing* dilakukan untuk melakukan pemotongan kalimat-kalimat dalam dokumen menjadi kata-kata penyusun kalimat. Proses *case folding* adalah proses untuk mengganti huruf kapital menjadi huruf kecil. Proses filtering merupakan proses pengambilan kata dengan menggunakan cara *stopword*. Proses *stemming* adalah proses penemukan akar kata dari hasil filtering. Kemudian, menghitung pembobotan kata menggunakan metode TF-IDF. Langkah selanjutnya, melakukan proses pendeteksian pada teks dengan Metode *Latent Semantics Analysis* (LSA).

Metode *Latent Semantics Analysis* (LSA) ditunjukkan oleh Gambar 3.a), terdiri dari dua langkah yaitu pertama adalah Singular Value Decomposition, dimana alur sub proses Singular

value decomposition ditunjukkan pada Gambar 3.b) dan kedua, *Consine Similarity Measurement*. Terakhir, didapatkan nilai bobot kemiripan setiap jurnal yang telah diinputkan.



Gambar 3. a) Alur proses LSA; b) Alur proses detail *Singular Value Decomposition*

HASIL DAN PEMBAHASAN

Pada deteksi plagiarisme dengan metode *Latent Semantic Analysis* (LSA) bisa dilihat dengan tingkat akurasi / kebenaran. Pengujian dilakukan dengan pembagian dataset menjadi dua bagian, yaitu yang pertama data untuk training dengan jumlah 200 data, sednagkan yang kedua data untuk testing sebanyak 20 data. Data training adalah beberapa data yang digunakan untuk inisialisasi nilai probabilitas awal. Data testing adalah beberapa data hasil backup secara berkala oleh aplikasi yang kemudian akan diproses berdasar training data jurnal sebelumnya. Dimana pengujian dilakukan menggunakan metode pengujian *confusion matrix*. Pengujian dilakukan dengan menguji data jurnal sebanyak 20 data pada proses testing data.

Tabel 1. Hasil testing data deteksi plagiarism artikel jurnal

No	Artikel ID	Similarity index	Jumlah kalimat yang terdeteksi plagiarisme		Prosentase akurasi (%)
			Pada sistem	sebenarnya	
1	RPL_01	18.57	2	1	50
2	AI_01	17.54	2	2	100
3	AI_02	18.96	4	4	100
4	KBJ_01	17.62	10	10	100
5	AI_03	16.91	4	4	100
6	RPL_02	18.08	2	1	50
7	RPL_03	17.86	3	2	66.67
8	RPL_04	17.41	4	4	100
9	RPL_05	17.55	1	1	100
10	KBJ_02	17.47	0	0	100
11	AI_04	17.93	1	1	100
12	KBJ_03	17.19	2	2	100
13	KBJ_04	18.06	0	0	0
14	RPL_06	18.67	2	2	100
15	KBJ_05	16.95	5	5	100
16	KBJ_06	17.11	0	0	100
17	KBJ_07	17.44	11	10	90.91
18	AI_05	17.09	6	6	100
19	KBJ_08	18.66	1	1	100
20	KBJ_09	16.6	0	0	100
				Rata-rata	87.88

Tabel 1 merupakan hasil testing data deteksi plagiarisme jurnal pada 20 data. Dimana tabel terdiri dari 5 baris yaitu Nomer, Nama Jurnal, Jumlah Kalimat Yang Terdeteksi Plagiarisme, *Similarity Index*, dan Persentase Akurasi. Dimana persentase akurasi diperoleh dari perhitungan nilai jumlah kalimat yang benar terdeteksi plagiat dibagi nilai jumlah kalimat yang terdeteksi plagiat pada sistem dikali 100%. Dari perhitungan persentase akurasi yang telah dilakukan maka didapatkan rata – rata persentase akurasi sebesar 87.88 %. Persentase akurasi yang didapatkan cukup besar sehingga dapat dikatakan bahwa sistem yang dibuat cukup bagus. kesalahan deteksi plagiarisme dikarenakaan kata – kata yang digunakan dalam suatu kalimat memiliki nilai *similarity* yang tinggi walaupun dengan susunan kata yang berbeda. Padahal *paraphrase* pada kalimat tidak termasuk plagiarisme.

KESIMPULAN

Berdasarkan hasil pengujian dalam penelitian ini, dapat diambil simpulkan yaitu: metode *Latent Semantic Analysis* (LSA) dapat diimplementasikan untuk mendeteksi plagiarisme artikel jurnal. Dan berdasarkan uji coba pada sistem yang dikembangkan, kesalahan deteksi plagiarisme dikarenakaan kata – kata yang digunakan dalam suatu kalimat memiliki nilai similarity yang tinggi walaupun dengan susunan kata yang berbeda, padahal paraphrase pada kalimat tidak termasuk plagiarisme.

DAFTAR PUSTAKA

[1] N. Nurdin, R. Rizal, and R. Rizwan, "Pendeteksian Dokumen Plagiarisme dengan Menggunakan Metode Weight Tree," *Telematika*, vol. 12, no. 1, p. 31, 2019, doi:

10.35671/telematika.v12i1.775.

[2] I. Nawawi, P. P. Arhandi, and F. Rahutomo, "DETEKSI PLAGIARISME PADA DOKUMEN SKRIPSI BERDASARKAN TINGKAT KESAMAAN DENGAN MENGGUNAKAN METODE LONGEST COMMON SUBSEQUENCE," *Janapati*, vol. 8, no. 3, pp. 217–226, 2019.

[3] R. P. Pratama, M. Faisal, and A. Hanani, "Deteksi Plagiarisme pada Dokumen Jurnal Menggunakan Metode Cosine Similarity," *SMARTICS J.*, vol. 5, no. 1, pp. 22–26, 2019, doi: 10.21067/smartics.v5i1.2848.

[4] I. Azharyani and D. Sulistyo Kusumo, "Implementasi Semantic Search pada Open Library menggunakan Metode Latent Semantic Analysis (Studi Kasus: Open Library Universitas Telkom)," *Agustus*, vol. 6, no. 2, p. 8987, 2019.

[5] H. Jayadianti, R. Damayanti, and ..., "Latent Semantic Analysis (Lsa) Dan Automatic Text Summarization (Ats) Dalam Optimasi Pencarian Artikel Covid 19," in *Seminar Nasional ...*, 2020, pp. 52–59. [Online]. Available: http://jurnal.upnyk.ac.id/index.php/semnasif/article/view/4085

[6] A. Hermawan, "Kebijakan Dosen Mengurangi Plagiarisme pada Karya Ilmiah Mahasiswa," *IJIP Indones. J. Islam. Psychol.*, vol. 1, no. 2, pp. 264–284, 2019, doi: 10.18326/ijip.v1i2.264-284.

[7] R. Siringoringo and J. Jamaludin, "Text Mining dan Klasterisasi Sentimen Pada Ulasan Produk Toko Online," *J. Teknol. dan Ilmu Komput. Prima*, vol. 2, no. 1, pp. 41–48, 2019, doi: 10.34012/jutikomp.v2i1.456.

[8] T. S. Kartikasari, H. Setiawan, and P. Lucky Tirma Irawan, "Implementasi Text Mining Untuk Analisis Opini Publik Terhadap Calon Presiden," *J. Simantec*, vol. 7, no. 1, pp. 39–47, 2020, doi: 10.21107/simantec.v7i1.6528.

[9] O. Berlin, D. S. Naga, and V. C. Mawardi, "Perancangan Aplikasi Pendeteksi Kemiripan Teks Dengan Menggunakan Metode Latent Semantic Analysis," *Comput. J. Comput. Sci. Inf. Syst.*, vol. 4, no. 1, p. 1, 2020, doi: 10.24912/computatio.v4i1.7191.