



Aplikasi Pendeteksi Plagiat Karya Tulis Ilmiah Berbasis Web Menggunakan NLP dan Algoritma KMP

Muhammad Alimu Rofiq^{1*}, Rizaldy Khair²

^{1*,2}Program Studi Sistem Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara, Medan, Indonesia

Email: ^{1*}alimuropiq969@gmail.com, ²rizaldykhair@umsu.ac.id

Abstrak

Plagiarisme merupakan permasalahan serius dalam dunia akademik yang dapat menurunkan integritas ilmiah dan kualitas karya tulis ilmiah. Pemeriksaan plagiarisme secara manual dinilai kurang efektif karena membutuhkan waktu lama dan mengandung unsur subjektivitas. Penelitian ini bertujuan merancang dan membangun aplikasi pendeteksi plagiat terhadap karya tulis ilmiah berbasis web dengan mengintegrasikan Natural Language Processing (NLP) dan algoritma Knuth-Morris-Pratt (KMP). NLP digunakan dalam tahapan preprocessing teks meliputi case folding, cleaning, tokenizing, stopword removal, dan stemming menggunakan algoritma Nazief-Adriani yang disesuaikan untuk morfologi Bahasa Indonesia. Algoritma KMP kemudian digunakan untuk melakukan pencocokan pola kata secara efisien antar dokumen. Metode penelitian yang digunakan adalah Research and Development (R&D). Sistem diimplementasikan menggunakan PHP native 8.1, MySQL, Tailwind CSS, dan Alpine.js. Pengujian fungsional menggunakan black box testing terhadap 20 skenario menunjukkan seluruh fitur sistem berjalan valid. Pengujian performa menunjukkan sistem mampu menangani corpus lebih dari 2.400 dokumen dengan 11 juta kata, dengan waktu eksekusi 2,93 detik pada end-to-end test simulasi 635 dokumen. Pengujian akurasi similarity menghasilkan nilai 81,97%, 74,07%, dan 10,42%, membuktikan sistem mampu membedakan secara konsisten dokumen dengan tingkat kemiripan tinggi dan rendah. Perbandingan dengan Turnitin menunjukkan selisih 13,38 poin persentase yang disebabkan perbedaan cakupan corpus, bukan kesalahan algoritma. Dengan demikian, aplikasi yang dibangun dapat membantu proses pendeteksian kemiripan karya tulis ilmiah berbahasa Indonesia secara efektif, transparan, dan bebas biaya lisensi.

Kata Kunci: Plagiarisme, Natural Language Processing, Knuth-Morris-Pratt, Similarity, Karya Tulis Ilmiah.

PENDAHULUAN

Perkembangan teknologi informasi pada era digital memberikan dampak signifikan terhadap dunia pendidikan, khususnya dalam kemudahan akses informasi melalui internet. Mahasiswa maupun penulis dapat memperoleh berbagai referensi secara cepat untuk mendukung penyusunan karya tulis ilmiah. Namun, kemudahan akses tersebut juga meningkatkan risiko terjadinya plagiarisme pada tugas, artikel ilmiah, jurnal, maupun skripsi. Plagiarisme merupakan tindakan mengambil atau menggunakan sebagian maupun seluruh karya orang lain tanpa mencantumkan sumber secara tepat sehingga bertentangan dengan etika akademik dan hak kekayaan intelektual. Kondisi ini menjadi perhatian penting karena dapat memengaruhi kualitas akademik, integritas ilmiah, serta kredibilitas institusi pendidikan (Harahap, 2024).

Pemeriksaan plagiarisme secara manual dinilai kurang efektif karena membutuhkan waktu yang relatif lama, terutama apabila jumlah dokumen yang diperiksa cukup banyak. Selain itu, pemeriksaan manual berpotensi menghasilkan subjektivitas dalam menentukan tingkat kemiripan suatu dokumen. Darmanto et al. (2024) menunjukkan bahwa penerapan sistem deteksi plagiarisme berbasis Natural Language Processing (NLP) menggunakan algoritma Jaro-Winkler dan TF-IDF mampu membantu meningkatkan efektivitas analisis kemiripan dokumen dibandingkan pemeriksaan manual.

Saat ini, sistem yang banyak digunakan dalam lingkungan akademik untuk mendeteksi plagiarisme adalah Turnitin. Akan tetapi, penggunaan sistem tersebut memiliki keterbatasan, yakni bersifat berbayar sehingga tidak seluruh mahasiswa maupun institusi pendidikan dapat menggunakannya secara mandiri. Pengembangan sistem deteksi plagiarisme berbasis web menjadi kebutuhan penting bagi institusi pendidikan sebagai alternatif dari sistem berbayar (Pratama et al., 2023).

Salah satu teknologi yang dapat diterapkan dalam sistem pendeteksi plagiarisme adalah Natural Language Processing (NLP). Putra dan Lestari (2023) menyatakan bahwa penerapan NLP dalam sistem deteksi plagiarisme dapat meningkatkan akurasi identifikasi kemiripan teks dibandingkan metode tradisional berbasis string matching. NLP merupakan cabang kecerdasan buatan yang memungkinkan komputer memahami, memproses, dan menganalisis bahasa alami manusia, mencakup tahapan case folding, tokenizing, filtering, dan stemming. Salman (2025) menunjukkan bahwa penerapan NLP pada sistem pendeteksi kemiripan judul skripsi berbasis web mampu mengenali pola kesamaan teks secara lebih efektif.

Selain NLP, algoritma Knuth-Morris-Pratt (KMP) memiliki peran penting dalam pendeteksian kemiripan teks. Algoritma ini bekerja dengan mencocokkan pola string secara efisien tanpa melakukan pengulangan pemeriksaan karakter yang tidak diperlukan. Herlina dan Rahmi (2025) menunjukkan bahwa algoritma KMP dapat diterapkan untuk mendeteksi tingkat plagiarisme pada tugas mahasiswa secara efektif. Islamiyati dan Fikri (2022) juga membuktikan bahwa algoritma KMP mampu diterapkan dalam sistem pendeteksian kemiripan judul skripsi berbasis web dengan proses pencarian teks yang efisien.

Penelitian-penelitian terdahulu, seperti penggunaan TF-IDF dan Cosine Similarity (Afandi, 2024; Halim, 2024; Mawanta et al., 2021), algoritma Levenshtein Distance (Setiawan & Saprudin, 2023), Rabin-Karp (Zidan, 2025), maupun pendekatan NLP murni (Salman, 2025; Akbar, 2025), masih menerapkan metode secara terpisah dan lebih banyak berfokus pada pendeteksian judul skripsi. Penelitian yang secara khusus mengintegrasikan preprocessing berbasis NLP dengan algoritma KMP pada aplikasi pendeteksi plagiarisme terhadap karya tulis ilmiah berbasis web masih terbatas. Kondisi tersebut menunjukkan adanya research gap yang menjadi dasar penelitian ini.

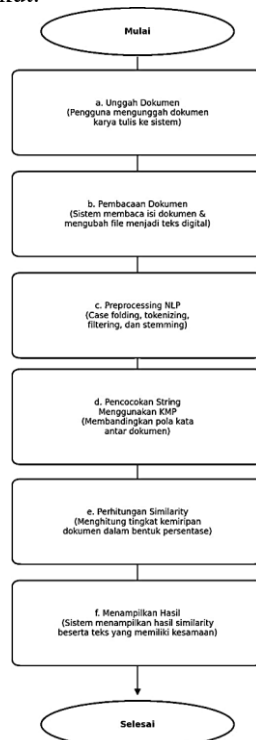
Berdasarkan hal tersebut, penelitian ini dilakukan dengan tujuan merancang, membangun, dan menerapkan Natural Language Processing (NLP) serta algoritma Knuth-Morris-Pratt (KMP) pada aplikasi pendeteksi plagiat terhadap karya tulis ilmiah berbasis web, sehingga sistem yang dihasilkan mampu melakukan pendeteksian tingkat kemiripan teks secara akurat.

METODE

Penelitian ini menggunakan metode Research and Development (R&D) karena tidak hanya bertujuan menganalisis permasalahan, tetapi juga menghasilkan produk berupa aplikasi pendeteksi plagiarisme karya tulis berbasis web. Sugiyono (2021) menyatakan bahwa metode R&D merupakan metode penelitian yang digunakan untuk menghasilkan suatu produk tertentu serta menguji efektivitasnya.

Algoritma Sistem

Sistem mengintegrasikan dua komponen utama: Natural Language Processing (NLP) untuk preprocessing teks, dan algoritma Knuth-Morris-Pratt (KMP) untuk pencocokan pola. Tahapan pemrosesan sistem digambarkan pada flowchart berikut.



Gambar 1. Flowchart Tahapan Algoritma Sistem

Alur pemrosesan sistem terdiri atas enam tahap: (1) pengguna mengunggah dokumen karya tulis; (2) sistem membaca isi dokumen dan mengubah file menjadi teks digital; (3) sistem melakukan preprocessing NLP; (4) sistem melakukan pencocokan string menggunakan algoritma KMP; (5) sistem menghitung tingkat kemiripan; dan (6) sistem menampilkan hasil similarity beserta teks yang memiliki kesamaan.

Tahapan Preprocessing NLP

Dalam proses pengolahan teks berbahasa Indonesia, tahapan preprocessing NLP memiliki peran penting untuk meningkatkan kualitas data teks sebelum dilakukan analisis lebih lanjut (Ramadhani & Mulyanto, 2023). Tahapan yang diterapkan meliputi:

1. Case folding — menyeragamkan seluruh huruf menjadi huruf kecil agar sistem tidak membedakan kapitalisasi dalam proses pencocokan.
2. Cleaning — membersihkan teks dari URL, angka, tanda baca, dan simbol yang tidak relevan.
3. Tokenizing — memecah teks menjadi satuan kata (token) berdasarkan spasi. Proses tokenisasi dirumuskan sebagai $\text{Token}(T) = \{t_1, t_2, t_3, \dots, t_n\}$ (Fadillah et al., 2024), di mana T adalah dokumen teks, t adalah token kata, dan n adalah jumlah token.
4. Stopword removal — membuang kata-kata umum berbahasa Indonesia yang tidak memiliki makna pembeda signifikan (misalnya "yang", "dan", "di", "untuk").
5. Stemming — mengubah kata berimbuhan menjadi kata dasar menggunakan algoritma Nazief-Adriani yang disesuaikan untuk morfologi Bahasa Indonesia (Arifin, 2022).

Algoritma Knuth-Morris-Pratt (KMP)

Wulandari dan Hidayat (2022) menemukan bahwa KMP lebih unggul dalam kecepatan pencocokan pada dokumen teks berukuran besar dibandingkan Rabin-Karp. KMP bekerja dalam dua tahap: (1) pembentukan tabel prefix/Longest Prefix Suffix (LPS) yang menyimpan panjang awalan terpanjang dari pola yang juga merupakan akhiran, dan (2) pencarian pola menggunakan tabel LPS sehingga algoritma dapat melompati pemeriksaan karakter yang sudah dipastikan cocok ketika terjadi ketidakcocokan. Kompleksitas waktu algoritma KMP dinyatakan sebagai $O(n + m)$, dengan n adalah panjang teks dan m adalah panjang pola (Ananditya et al., 2022).

Rumus Similarity

Similarity dokumen dihitung menggunakan persamaan berikut:

$$\text{Similarity} = \left(\frac{\text{Jumlah Kata Sama}}{\text{Total Kata}} \right) \times 100\% \quad (1)$$

di mana Jumlah Kata Sama adalah jumlah kata unik hasil preprocessing NLP pada dokumen pengguna yang ditemukan pada dokumen pembandingan melalui pencarian pola KMP, dan Total Kata adalah jumlah kata unik gabungan kedua dokumen (Islamiyati & Fikri, 2022).

Perancangan UML

Perancangan sistem menggunakan Unified Modeling Language (UML) yang terdiri atas Use Case Diagram, Activity Diagram, Sequence Diagram, dan Class Diagram (Saputra, 2024). Sistem memiliki dua aktor: Pengguna (user) yang dapat melakukan deteksi plagiarisme dan mengelola riwayat pemeriksaan, serta Admin yang memiliki seluruh hak akses Pengguna ditambah kewenangan mengelola corpus dan pengaturan sistem.



Gambar 2. Use Case Diagram Sistem

Implementasi Sistem

Sistem SimilarCheck diimplementasikan menggunakan PHP native versi 8.1, MySQL sebagai sistem manajemen basis data, dengan antarmuka pengguna dibangun menggunakan Tailwind CSS dan Alpine.js. Basis data sistem terdiri atas lima tabel utama yang saling berelasi: users, dokumen, pemeriksaan, pemeriksaan_detail, dan pengaturan.

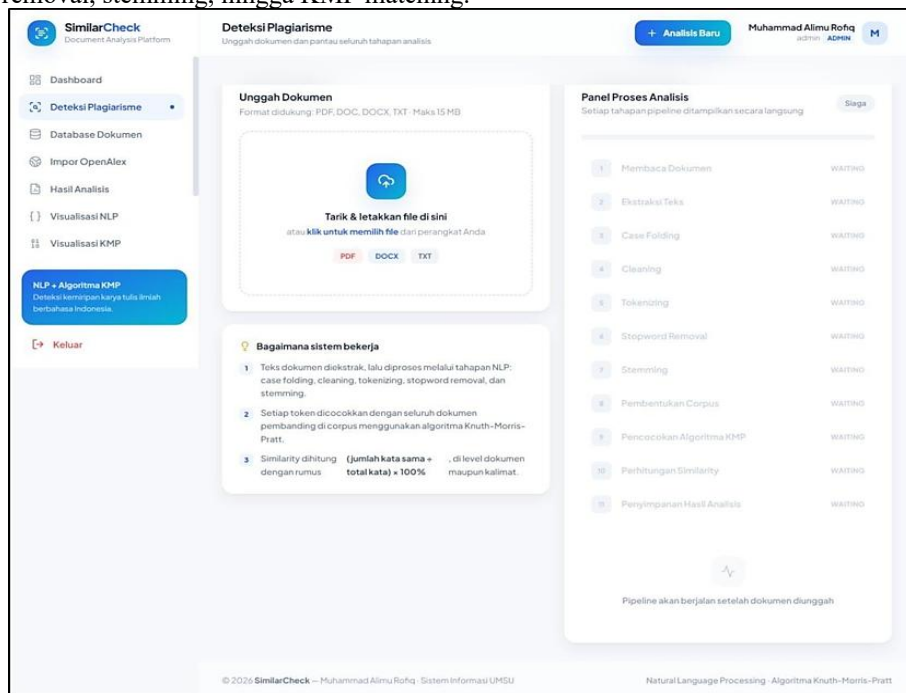
Pengujian

Pengujian dilakukan dalam tiga tahap: (1) pengujian fungsional menggunakan black box testing terhadap 20 skenario untuk memverifikasi seluruh fungsi sistem (Permana, 2023); (2) pengujian performa sistem untuk memastikan sistem tetap responsif pada corpus berskala besar; dan (3) pengujian akurasi similarity menggunakan tiga berkas uji yang mewakili Bab Pendahuluan, Landasan Teori, dan Metodologi.

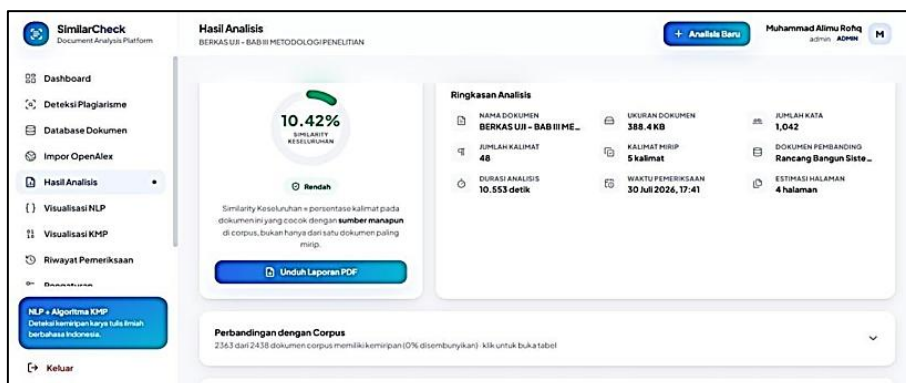
HASIL DAN PEMBAHASAN

Implementasi Antarmuka Sistem

Sistem SimilarCheck dibangun dengan antarmuka yang mencakup seluruh fitur yang dibutuhkan dalam proses deteksi plagiarisme. Halaman utama sistem memungkinkan pengguna mengunggah dokumen dan memantau progres analisis secara real-time, mencakup indikator tahapan case folding, tokenizing, stopword removal, stemming, hingga KMP matching.

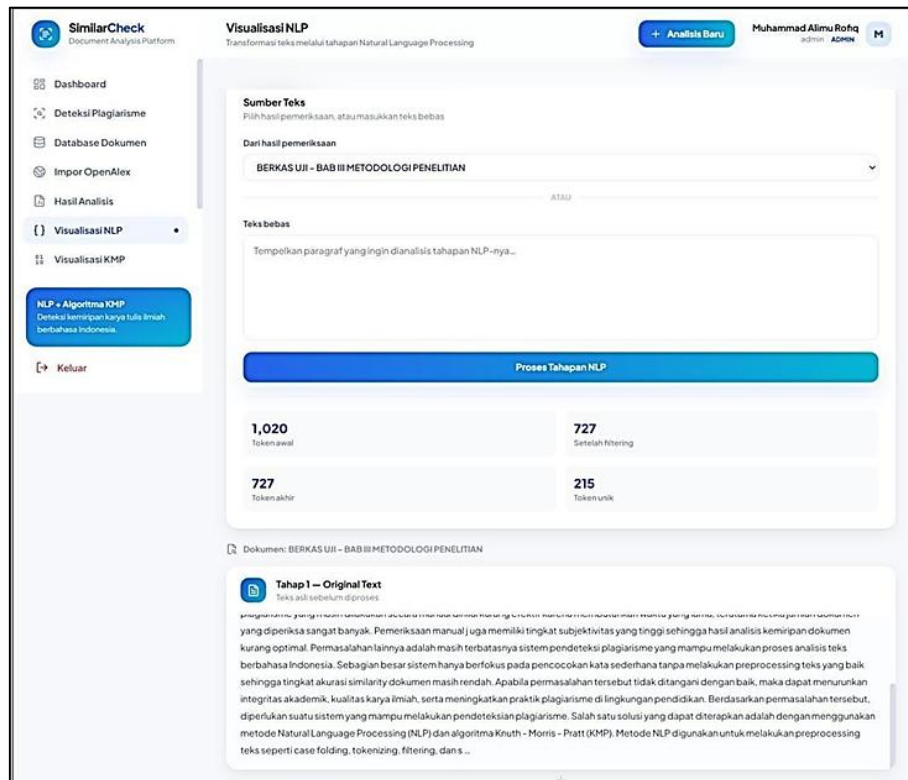


Gambar 3. Halaman Deteksi Plagiarisme

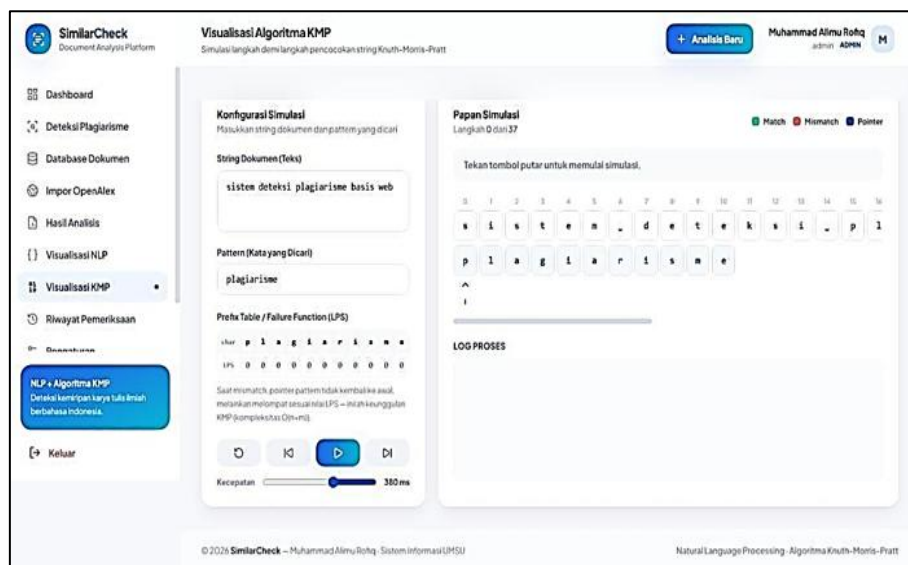


Gambar 4. Halaman Hasil Analisis Similarity

Salah satu keunggulan sistem adalah tersedianya halaman Visualisasi NLP yang menampilkan setiap tahapan preprocessing secara transparan, serta halaman Visualisasi KMP yang mendemonstrasikan cara kerja algoritma KMP secara bertahap, termasuk pembentukan tabel prefix (LPS) dan proses pergeseran pola saat pencocokan karakter — sebuah fitur yang tidak ditemukan pada sistem pendeteksi plagiarisme komersial umumnya.



Gambar 5. Halaman Visualisasi Tahapan NLP



Gambar 6. Halaman Visualisasi Algoritma KMP

Implementasi Algoritma KMP

Sebagai ilustrasi cara kerja algoritma KMP dalam sistem, perhatikan pencarian kata dasar "cocok" pada dokumen. Tabel 1 menunjukkan tabel LPS yang terbentuk untuk pola "cocok":

Tabel 1. Tabel Prefix (LPS) untuk Pola "cocok"

Indeks (i)	0	1	2	3	4
Karakter Pola	c	o	c	o	k
Nilai LPS	0	0	1	2	0

Dengan tabel LPS ini, ketika terjadi ketidakcocokan pada posisi tertentu, algoritma langsung melompati pemeriksaan ke posisi yang ditentukan tabel, tanpa mengulang dari awal — menjamin kompleksitas waktu $O(n + m)$ yang jauh lebih efisien dibandingkan pendekatan brute-force.

Pengujian Fungsional (Black Box Testing)

Pengujian fungsional dilakukan terhadap 20 skenario penggunaan yang mencakup seluruh fitur utama sistem. Hasil pengujian disajikan pada Tabel 2.

Tabel 2. Hasil Pengujian Fungsional (Black Box Testing)

No.	Skenario Pengujian	Hasil yang Diharapkan	Kesimpulan
1	Login dengan username dan password benar	Masuk ke Dashboard (admin) / Deteksi (user)	Valid
2	Login dengan password salah	Menampilkan pesan gagal login	Valid
3	Registrasi akun baru sebagai user	Akun tersimpan dengan role 'user'	Valid
4	User biasa mengakses menu admin via URL	Sistem menolak / redirect ke halaman Deteksi	Valid
5	Admin mengunggah dokumen ke corpus (format PDF)	Dokumen tersimpan dan diproses NLP otomatis	Valid
6	Admin mengunggah banyak dokumen sekaligus (format campuran)	Seluruh dokumen valid tersimpan; ringkasan per berkas ditampilkan	Valid
7	Mengunggah dokumen PDF yang sulit diekstrak	Sistem otomatis mencoba metode ekstraksi cadangan	Valid
8	Mengunggah berkas berformat tidak didukung	Sistem menolak berkas dengan pesan jelas	Valid
9	Menghapus dokumen dari corpus	Dokumen terhapus dari database	Valid
10	Menjalankan deteksi pada dokumen bersimilaritas tinggi	Similarity terhitung dan kalimat mirip ter-highlight	Valid
11	Menjalankan deteksi pada dokumen tanpa kemiripan	Similarity 0% dan tidak ada kalimat ter-highlight	Valid
12	User A mencoba melihat hasil analisis milik User B	Sistem menolak akses (bukan pemilik data)	Valid
13	Menghapus riwayat pemeriksaan milik sendiri	Riwayat dan detail terkait terhapus dari sistem	Valid
14	Menghapus riwayat milik pengguna lain	Sistem menolak permintaan	Valid
15	Mencari karya ilmiah lewat menu Impor OpenAlex	Daftar hasil pencarian berteks lengkap ditampilkan	Valid
16	Mengimpor karya terpilih dari OpenAlex ke corpus	Dokumen tersimpan dengan teks lengkap hasil ekstraksi	Valid
17	Mengubah ambang batas similarity di Pengaturan	Kategori hasil analisis berikutnya mengikuti ambang baru	Valid
18	Mengunduh laporan hasil analisis format PDF	Berkas PDF terunduh berisi ringkasan dan rincian hasil	Valid
19	Membuka halaman Visualisasi KMP	Tahapan pencocokan pola ditampilkan langkah demi langkah	Valid
20	Mengakses halaman admin tanpa login	Sistem redirect ke halaman Login	Valid

Berdasarkan Tabel 2, seluruh 20 skenario pengujian fungsional dinyatakan valid, yang berarti sistem telah berjalan sesuai rancangan dan kebutuhan fungsional yang ditetapkan.

Pengujian Performa Sistem

Pengujian performa dilakukan untuk memastikan sistem tetap responsif pada corpus berskala besar. Pada implementasi produksi aktual, corpus telah mencapai 2.438 dokumen dengan total 11.169.879 kata yang telah dianalisis secara kumulatif.

Optimasi pertama dilakukan pada penggunaan memori. Pendekatan awal yang memuat seluruh corpus ke memori sekaligus terbukti tidak efisien. Sistem dioptimasi dengan pendekatan pemrosesan bertahap (streaming), hasilnya disajikan pada Tabel 3.

Tabel 3. Perbandingan Penggunaan Memori Sebelum dan Sesudah Optimasi

Pendekatan	Penggunaan Memori Puncak	Keterangan
Sebelum optimasi (muat seluruh corpus sekaligus)	281,1 MB	Berisiko melebihi batas memory_limit peladen pada corpus besar
Sesudah optimasi (pemrosesan bertahap/streaming)	1,7 MB	Penurunan ≈ 165 kali lipat, stabil untuk corpus berapapun besarnya

Optimasi kedua dilakukan pada waktu eksekusi menggunakan pendekatan inverted index dan filter matematis untuk mempersempit kandidat kalimat yang perlu dibandingkan, hasilnya disajikan pada Tabel 4 berikut.

Tabel 4. Perbandingan Waktu Eksekusi Pencocokan Level-Kalimat

Pendekatan	Waktu Eksekusi	Keterangan
Sebelum optimasi (brute-force)	> 250 detik	Melebihi batas waktu eksekusi peladen (timeout)
Sesudah optimasi (inverted index + filter matematis)	17,65 detik	Hasil similarity identik, waktu turun drastis

Pada pengujian end-to-end dengan simulasi corpus 635 dokumen ($\approx 4,9$ juta kata), keseluruhan proses deteksi dari ekstraksi teks hingga penyimpanan hasil akhir terselesaikan dalam 2,93 detik.

Pengujian Akurasi Similarity

Pengujian akurasi dilakukan menggunakan tiga berkas uji yang masing-masing mewakili Bab Pendahuluan, Bab Landasan Teori, dan Bab Metodologi. Hasilnya disajikan pada Tabel 5.

Tabel 5. Hasil Pengujian Akurasi Similarity

No.	Berkas Uji	Jumlah Kata	Jumlah Kalimat	Kalimat Mirip	Similarity (%)	Kategori
1	Bab I Pendahuluan	1.471	61	50	81,97	Tinggi
2	Bab II Landasan Teori	3.649	162	120	74,07	Tinggi
3	Bab III Metodologi	1.042	48	5	10,42	Rendah

Hasil pada Tabel 5 menunjukkan sistem mampu membedakan secara konsisten antara dokumen dengan tingkat kemiripan tinggi dan dokumen dengan tingkat kemiripan rendah. Nilai similarity yang tinggi pada dua berkas uji pertama sejalan dengan proporsi kalimat mirip yang juga tinggi (82% dan 74%), sedangkan nilai similarity yang rendah pada berkas uji ketiga sejalan dengan proporsi kalimat mirip yang jauh lebih kecil (10%).

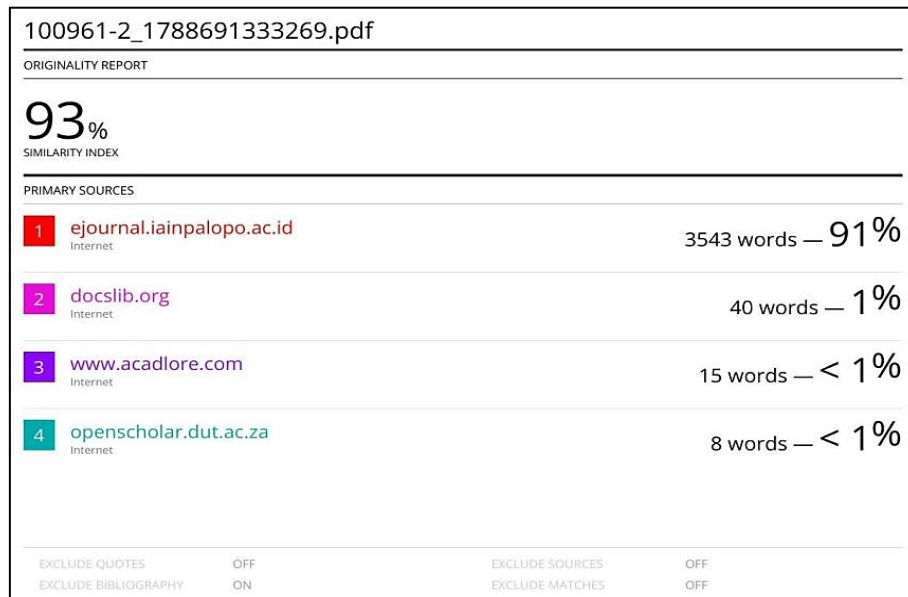
Perbandingan dengan Turnitin

Untuk memvalidasi kewajaran hasil, dilakukan perbandingan terhadap Turnitin menggunakan dokumen uji artikel ilmiah yang telah dipublikasikan terbuka. Hasil perbandingan disajikan pada Tabel 6.

Tabel 6. Perbandingan Hasil Deteksi SimilarCheck dengan Turnitin

Aspek	Turnitin	SimilarCheck
Similarity Keseluruhan	93%	79,62%
Sumber Kemiripan Tertinggi	ejournal.iainpalopo.ac.id (91%)	Artikel bertema e-learning sejenis (21,54%)
Cakupan Basis Data	Indeks jurnal dan repositori global	Corpus internal (lokal)
Metode Deteksi	Bersifat tertutup (proprietary)	NLP + algoritma KMP (terbuka/transparan)
Transparansi Proses	Tidak ditampilkan kepada pengguna	Dapat ditelusuri melalui halaman Visualisasi
Biaya/Lisensi	Berlangganan (komersial)	Mandiri (self-hosted), bebas biaya lisensi

Terdapat selisih similarity sebesar 13,38 poin persentase. Selisih ini bukan disebabkan kesalahan pada tahapan NLP maupun algoritma KMP, melainkan oleh perbedaan cakupan corpus — Turnitin memiliki akses ke basis data global yang mencakup versi terbitan asli dokumen uji, sedangkan corpus SimilarCheck pada saat pengujian belum memuat dokumen sumber tersebut. Temuan ini sekaligus membuktikan bahwa metode dan algoritma yang diterapkan bekerja secara konsisten, karena seluruh 79,62% kemiripan yang dilaporkan merupakan kecocokan kalimat yang dapat diverifikasi pada corpus yang tersedia.



Gambar 7. Hasil Deteksi Similarity oleh Turnitin



Gambar 8. Hasil Deteksi Similarity oleh SimilarCheck

Kelebihan dan Keterbatasan Sistem

Sistem SimilarCheck memiliki beberapa kelebihan utama: (1) transparansi proses — seluruh tahapan NLP dan algoritma KMP dapat ditelusuri langsung oleh pengguna, berbeda dengan sistem komersial yang bersifat black box; (2) dukungan khusus Bahasa Indonesia melalui stemming Nazief-Adriani; (3) bersifat mandiri (self-hosted) dan bebas biaya lisensi; (4) skalabilitas corpus yang terbukti mampu menangani lebih dari 2.400 dokumen dan 11 juta kata tanpa kegagalan memori; serta (5) integrasi otomatis dengan OpenAlex untuk memperkaya corpus secara berkelanjutan.

Di sisi keterbatasan, sistem memiliki cakupan corpus yang bergantung sepenuhnya pada dokumen yang diunggah atau diimpor. Pencocokan berbasis kata hasil KMP juga bersifat sensitif terhadap kesamaan kata eksak, sehingga kurang peka terhadap plagiarisme berbentuk parafrasa. Sistem belum mendukung deteksi kemiripan lintas Bahasa.

KESIMPULAN

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan, dapat ditarik kesimpulan sebagai berikut:

1. Aplikasi pendeteksi plagiat terhadap karya tulis ilmiah berbasis web telah berhasil dirancang dan dibangun menggunakan Natural Language Processing (NLP) dan algoritma Knuth-Morris-Pratt (KMP). Sistem diimplementasikan menggunakan PHP native 8.1 dan MySQL dengan lima tabel basis

- data yang saling berelasi, mencakup fitur autentikasi dua peran pengguna, pengelolaan corpus, deteksi plagiarisme, visualisasi proses NLP dan KMP, riwayat pemeriksaan, hingga pencetakan laporan PDF.
2. Tahapan NLP yang diterapkan — meliputi case folding, cleaning, tokenizing, stopword removal, dan stemming menggunakan algoritma Nazief-Adriani — serta algoritma KMP dengan pembentukan tabel prefix (LPS) telah berhasil diterapkan dalam proses pendeteksian kemiripan teks pada dokumen karya tulis ilmiah berbahasa Indonesia.
 3. Hasil pengujian membuktikan sistem berfungsi dengan baik. Seluruh 20 skenario black box testing dinyatakan valid. Pengujian performa menunjukkan sistem mampu menangani corpus lebih dari 2.400 dokumen dengan 11 juta kata secara stabil, dengan waktu eksekusi end-to-end 2,93 detik pada simulasi 635 dokumen. Pengujian akurasi menunjukkan sistem mampu membedakan secara konsisten dokumen dengan tingkat kemiripan tinggi (81,97% dan 74,07%) dan rendah (10,42%).

Untuk penelitian selanjutnya, disarankan untuk mengintegrasikan lebih banyak sumber basis data karya ilmiah terbuka, mengombinasikan algoritma KMP dengan analisis kemiripan semantik berbasis word embedding untuk mendeteksi parafrasa, mengembangkan mekanisme ekstraksi OCR untuk dokumen PDF hasil pindaian, serta menambahkan dukungan deteksi kemiripan lintas bahasa.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Universitas Muhammadiyah Sumatera Utara, khususnya Program Studi Sistem Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, atas dukungan dalam pelaksanaan penelitian ini.

DAFTAR PUSTAKA

- Afandi, M. D. (2024). Penerapan Information Retrieval dalam Sistem Analisis Kemiripan Proposal Skripsi menggunakan Cosine Similarity. *SWABUMI*. <https://ojs.bsi.ac.id/index.php/swabumi/article/view/17087>
- Akbar. (2025). Identification of Source Code Plagiarism Using a Natural Language Processing Approach Based on Code Writing Style Analysis. *Journal of Applied Intelligent Computing*. <https://doi.org/10.30871/jaic.v9i6.11206>
- Ananditya, A., Setyawan, A., & Wahono, R. S. (2022). Perbandingan Algoritma String Matching: Brute Force, Knuth-Morris-Pratt, dan Boyer-Moore dalam Pencarian Teks. *Scientific Journal of Informatics*, 9(1). <https://doi.org/10.15294/sji.v9i1.35580>
- Arifin, A. Z. (2022). Enhanced Nazief-Adriani Stemmer for Indonesian Text Processing. *International Journal of Intelligent Engineering and Systems*. <https://doi.org/10.22266/ijies2022.0630.34>
- Darmanto, D., Pradasari, N. I., & Wahyudi, E. (2024). Sistem deteksi plagiarisme tugas akhir mahasiswa berbasis Natural Language Processing menggunakan algoritma Jaro-Winkler dan TF-IDF. *Smart Comp: Jurnalnya Orang Pintar Komputer*, 13(1). <https://doi.org/10.30591/smartcomp.v13i1.6375>
- Fadillah, R. (2024). Implementasi Tokenisasi dan Stopword Removal pada Sistem Analisis Teks Bahasa Indonesia. *KOMIK*, 8(1). <https://doi.org/10.30865/komik.v8i1.7145>
- Halim. (2024). Aplikasi Deteksi Plagiarisme Dokumen dengan TF-IDF dan Cosine Similarity Berbasis Web. *Jurnal Bianglala Informatika*. <https://jurnal.kdi.or.id/index.php/bt/article/view/1697>
- Harahap, M. K. (2024). Analisis Tingkat Plagiarisme pada Karya Tulis Ilmiah Mahasiswa di Perguruan Tinggi. *JPDK*, 6(1). <https://doi.org/10.31004/jpdk.v6i1.12567>
- Haruna. (2024). Deteksi Tingkat Kemiripan Judul Skripsi Prodi Informatika Menggunakan Algoritma String Matching. *AJST*. <https://jurnal.ardenjaya.com/index.php/ajst/article/view/655>
- Herlina, E., & Rahmi, E. (2025). Penerapan Algoritma Knuth Morris Pratt untuk Mendeteksi Tingkat Plagiarisme pada Tugas Mahasiswa. *Garuda Kemdiktisaintek*. <https://garuda.kemdiktisaintek.go.id/documents/detail/5982709>
- Islamiyati, D. S., & Fikri, A. (2022). Penerapan algoritma Knuth-Morris-Pratt dalam mendeteksi tingkat kemiripan judul skripsi berbasis web. *Journal of Information System Research (JOSH)*, 3(2). <https://doi.org/10.47065/josh.v3i2.1168>
- Mawanta, dkk. (2021). Uji Kemiripan Kalimat Judul Tugas Akhir dengan Metode Cosine Similarity dan Pembobotan TF-IDF. *MIB*. <https://ejurnal.stmik-budidarma.ac.id/index.php/mib/article/view/2935>
- Nugroho, K. A. (2022). Analisis Perbandingan Metode Pengukuran Similarity Dokumen Teks Berbahasa Indonesia. *JITI*, 11(1). <https://doi.org/10.35457/jiti.v11i1.2158>
- Permana, A. A. (2023). Implementasi Black Box Testing pada Pengujian Aplikasi Berbasis Web. *JUTIF*, 4(3). <https://doi.org/10.20884/1.jutif.2023.4.3.896>
- Pratama, R. (2023). Pengembangan Sistem Deteksi Plagiarisme Karya Ilmiah Berbasis Web sebagai Alternatif Pemeriksaan Dokumen Akademik. *JSI*, 15(2). <https://doi.org/10.36706/jsi.v15i2.896>

- Putra, & Lestari. (2023). Penerapan Natural Language Processing dalam Sistem Deteksi Plagiarisme. *Jurnal Sistem Informasi dan Teknologi*. [Masukkan DOI/URL jika tersedia]
- Ramadhani, F. (2023). Implementasi Text Preprocessing Menggunakan Natural Language Processing untuk Analisis Teks Bahasa Indonesia. *JTIK*. <https://doi.org/10.25126/jtiik.2023102345>
- Salman. (2025). Sistem Pendeteksi Kemiripan Judul Skripsi Berbasis Web Menggunakan NLP. *Jurnal Penerbit Widina*. <https://jurnal.penerbitwidina.com/index.php/JPI/article/view/1964>
- Saputra, R. (2024). Penerapan Unified Modeling Language (UML) dalam Perancangan Sistem Informasi Berbasis Web. *RESTI*, 8(1). <https://doi.org/10.29207/resti.v8i1.5234>
- Setiawan, & Saprudin. (2023). Implementasi Sistem Pendeteksian Kemiripan Judul Skripsi dengan Algoritma Levenshtein Distance. *LOGIC*. <https://www.journal.mediapublikasi.id/index.php/logic/article/view/2638>
- Sihombing, & Anggriana. (2024). Analisis Similarity Dokumen Menggunakan Natural Language Processing dan Cosine Similarity. *Jurnal Informatika dan Rekayasa Perangkat Lunak*. [Masukkan DOI/URL jika tersedia]
- Sugiyono. (2021). *Metode Penelitian Kuantitatif, Kualitatif, dan R&D*. Alfabeta.
- Wahyudi, E. (2022). Penerapan Text Mining untuk Ekstraksi Informasi pada Dokumen Teks Berbahasa Indonesia. *TIKomsin*, 10(2). <https://doi.org/10.30646/tikomsin.v10i2.654>
- Wibowo, A. (2023). Pengembangan Aplikasi Berbasis Web Menggunakan Framework Modern untuk Sistem Informasi Akademik. *JIRPL*, 5(1). <https://doi.org/10.33365/jirpl.v5i1.2456>
- Wulandari, & Hidayat. (2022). Studi Perbandingan Algoritma Knuth-Morris-Pratt dan Rabin-Karp dalam Pencocokan String pada Dokumen Teks. *Jurnal Informatika Sunan Kalijaga*. [Masukkan DOI/URL jika tersedia]
- Zidan. (2025). Implementasi Algoritma Rabin-Karp dalam Pendeteksian Plagiarisme. *JIMIK*. <https://journal.stmiki.ac.id/index.php/jimik/article/view/1191>