

Optimalisasi Jaro-Winker Dan Levenshtein Untuk Sistem Penyaringan Judul Skripsi Di Fakultas Teknik UMB

Muhammad Adriansyah¹⁾; Dedy Abdullah²⁾; Ardi Wijaya³⁾; Gunawan⁴⁾
^{1,2,3,4)}Program Studi Teknik Informatika Fakultas Teknik, Universitas Muhammadiyah Bengkulu
Email: ¹⁾adriansyahm424@gmail.com

ARTICLE HISTORY

Received [29 Juli 2026]
Revised [20 Oktober 2026]
Accepted [21 Oktober 2026]

KEYWORDS

Jaro-Winkler, Levenshtein,
Threshold, Thesis.

This is an open access
article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/)
license



ABSTRAK

Penyaringan judul skripsi penting dilakukan untuk mencegah duplikasi karya ilmiah di Fakultas Teknik Universitas Muhammadiyah Bengkulu.² Penelitian ini menguji optimasi batas ambang kemiripan (*threshold similarity*) pada algoritma Jaro-Winkler dan Levenshtein Distance menggunakan 50 pasangan judul skripsi fiktif.³ Pengujian biner menunjukkan bahwa algoritma Jaro-Winkler mencapai kinerja optimal dengan *F1-score* sebesar 1,000 pada *threshold* 0,85.³ Sementara itu, Levenshtein Distance mencapai performa optimal pada *threshold* 0,70 dengan *F1-score* sebesar 0,837.³ Hasil uji Wilcoxon Signed-Rank membuktikan adanya perbedaan performa yang signifikan secara statistik antara kedua algoritma dengan nilai $p = 0,012 < 0,05$.

ABSTRACT

Thesis title screening is crucial to prevent duplication of scientific work in the Faculty of Engineering, Universitas Muhammadiyah Bengkulu.² This study examines the optimization of the similarity threshold for Jaro-Winkler and Levenshtein Distance algorithms using 50 dummy thesis title pairs.³ Binary evaluation shows that the Jaro-Winkler algorithm achieves optimal performance with an F1-score of 1.000 at a 0.85 threshold.³ Meanwhile, Levenshtein Distance achieves its optimal performance at a 0.70 threshold with an F1-score of 0.837.³ The Wilcoxon Signed-Rank test confirms a statistically significant difference in performance between the two algorithms with $p = 0.012 < 0.05$. The title screening system is recommended to implement the Jaro-Winkler algorithm.³

PENDAHULUAN

Dalam lingkungan akademik, terutama pada jenjang pendidikan tinggi, keberadaan sistem pengelolaan skripsi yang efektif dan efisien merupakan aspek penting untuk menjamin orisinalitas karya ilmiah mahasiswa. Berdasarkan data Direktorat Jenderal Pendidikan Tinggi (DIKTI, 2023), kasus kemiripan atau duplikasi judul tugas akhir meningkat seiring dengan pertumbuhan jumlah mahasiswa dan keterbatasan sistem pengawasan digital di kampus. Permasalahan ini tidak hanya menimbulkan hambatan administratif, tetapi juga berpotensi menurunkan kredibilitas dan kualitas penelitian apabila tidak terdeteksi secara dini (Rahman et al., 2022).

Penelitian ini berfokus pada optimasi *threshold similarity* untuk kedua algoritma tersebut agar dapat mencapai keseimbangan terbaik antara *precision* dan *recall* dalam konteks sistem penyaringan judul skripsi. Penelitian ini diharapkan dapat memberikan kontribusi terhadap pengembangan sistem deteksi kemiripan teks yang lebih adaptif dan akurat, khususnya di lingkungan perguruan tinggi. Dengan adanya optimasi *threshold* yang tepat, sistem penyaringan judul skripsi dapat membantu pihak administrasi dan dosen pembimbing dalam meminimalisasi kemiripan judul serta meningkatkan efisiensi proses pengajuan proposal penelitian (Hidayat et al., 2020).

Berdasarkan latar belakang tersebut, penelitian ini berupaya menjawab beberapa permasalahan utama. Pertama, bagaimana kinerja algoritma Jaro-Winkler dan Levenshtein dalam mendeteksi tingkat kemiripan judul skripsi di lingkungan FT UMB. Kedua, bagaimana pengaruh variasi nilai *threshold similarity* terhadap tingkat akurasi sistem dalam mengklasifikasikan judul yang mirip dan tidak mirip. Ketiga, bagaimana cara menentukan nilai *threshold similarity* yang optimal agar diperoleh keseimbangan terbaik antara nilai *precision* dan *recall* dalam sistem deteksi kemiripan judul skripsi.

Sejalan dengan rumusan masalah tersebut, tujuan penelitian ini adalah untuk menganalisis performa algoritma Jaro-Winkler dan Levenshtein Distance dalam mendeteksi kemiripan judul skripsi. Selain itu, penelitian ini bertujuan untuk mengetahui dampak variasi nilai *threshold similarity* terhadap hasil klasifikasi sistem. Tujuan akhir dari penelitian ini adalah menentukan nilai ambang (*threshold*) yang optimal sehingga sistem mampu menghasilkan kinerja terbaik, yang diukur berdasarkan nilai *F1-score* atau Youden's Index sebagai indikator keseimbangan antara *precision* dan *recall*.

TINJAUAN PUSTAKA

Konsep String Similarity

String similarity merupakan metode untuk mengukur tingkat kemiripan antara dua atau lebih string teks. Konsep ini penting dalam berbagai bidang seperti information retrieval, record linkage, dan deteksi plagiarisme (Christen, 2012). String similarity juga merupakan sebuah proses kuantitatif untuk mengukur tingkat kemiripan antara dua string berdasarkan perhitungan karakter atau pola tertentu, sehingga dua teks yang tidak identik dapat tetap dinilai mirip dalam konteks komputasi.

Algoritma Levenshtein Distance

Algoritma Levenshtein Distance diperkenalkan oleh Vladimir Levenshtein pada tahun 1966 untuk mengukur jarak antara dua string berdasarkan jumlah operasi edit (penambahan, penghapusan, atau penggantian karakter) yang diperlukan untuk mengubah satu string menjadi string lainnya (Levenshtein, 1966). Nilai jarak yang lebih kecil menunjukkan tingkat kemiripan yang lebih tinggi.

Algoritma Jaro–Winkler

Algoritma Jaro–Winkler dikembangkan oleh Matthew Winkler berdasarkan algoritma Jaro (Jaro, 1989; Winkler, 1999). Algoritma ini digunakan untuk menghitung kesamaan dua string dengan memperhatikan jumlah karakter yang cocok dan posisi relatifnya. Nilai kesamaan Jaro dihitung berdasarkan karakter yang cocok (*matching characters*) dan jumlah transpositions atau pertukaran posisi karakter.

Konsep Threshold Similarity

Threshold similarity adalah nilai ambang yang menentukan apakah dua string dianggap mirip atau tidak (Hidayat et al., 2020). Penentuan nilai threshold yang optimal menjadi faktor penting untuk menyeimbangkan antara *false positive* dan *false negative*. Nilai threshold yang terlalu tinggi dapat mengabaikan pasangan string yang sebenarnya mirip, sedangkan threshold yang terlalu rendah dapat menghasilkan terlalu banyak pasangan yang salah dikategorikan mirip (Maharani et al., 2020).

Sistem Informasi

Sistem informasi merupakan kombinasi antara teknologi, manusia, prosedur, serta basis data yang saling terintegrasi untuk mengolah data menjadi informasi yang berguna dalam mendukung proses operasional dan pengambilan keputusan pada suatu organisasi (Martono, 2022). Sistem informasi berfungsi untuk mengumpulkan, mengelola, menyimpan, dan mendistribusikan informasi sehingga dapat membantu meningkatkan efektivitas dan efisiensi kerja dalam suatu instansi atau perusahaan (Aryani et al., 2022).

METODE PENELITIAN

Jenis Penelitian

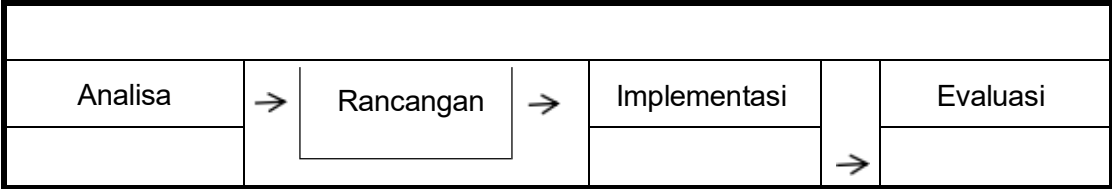
Penelitian ini merupakan penelitian kuantitatif dengan pendekatan eksperimen komparatif, yang bertujuan untuk membandingkan performa dua algoritma string similarity, yaitu Jaro–Winkler dan Levenshtein Distance, dalam mendeteksi kemiripan judul skripsi berdasarkan variasi nilai threshold. Metode eksperimen digunakan untuk mengetahui pengaruh perubahan parameter threshold terhadap tingkat akurasi sistem penyaringan judul (Creswell & Creswell, 2018). Pendekatan ini dipilih karena mampu memberikan hasil yang objektif melalui pengukuran numerik dan analisis statistik, sehingga hasil penelitian dapat diinterpretasikan secara terukur dan replikatif.

Sumber dan Jenis Data

Data yang digunakan dalam penelitian ini merupakan data sekunder simulatif (*dummy data*) yang disusun secara manual untuk keperluan pengujian sistem. Dataset terdiri dari sekitar 500 judul skripsi fiktif yang dibuat berdasarkan pola umum penulisan judul penelitian mahasiswa di bidang Informatika, Sistem Informasi, Manajemen, dan Pendidikan. Setiap judul diberi label “mirip” atau “tidak mirip” secara manual untuk membentuk *ground truth* yang akan digunakan dalam evaluasi performa algoritma. Labelisasi dilakukan berdasarkan penilaian kesamaan semantik dan sintaktik antarjudul, dengan mempertimbangkan kesamaan kata kunci utama dan struktur kalimat (Gomaa & Fahmy, 2013).

Tahapan Penelitian

Model pengembangan sistem yang digunakan dalam penelitian ini adalah model incremental, yang menekankan proses pengembangan perangkat lunak secara bertahap. Setiap tahapan menghasilkan bagian dari sistem yang berfungsi, dan pengembangan berlanjut hingga seluruh fungsi sistem selesai dan teruji.



Gambar 1. Diagram Incremental

Tahapan metode incremental dalam konteks penelitian ini terdiri atas:

- Analisa kebutuhan sistem dan data
- Perancangan algoritma dan struktur data
- Implementasi program dan uji coba algoritma
 - Evaluasi dan validasi hasil

Metode Penelitian

Tahap ini mencakup analisis kebutuhan data dan perancangan eksperimen untuk menguji efektivitas algoritma Jaro—Winkler dan Levenshtein dalam mendeteksi kemiripan judul skripsi. Adapun metode pengumpulan data yang dilakukan adalah:

Analisa kebutuhan sistem dan data

1. Dokumentasi:
Mengumpulkan data sekunder berupa kumpulan judul skripsi fiktif (dummy dataset) dari berbagai bidang studi, disusun berdasarkan pola umum penulisan judul penelitian di bidang Informatika.
2. Observasi literatur:
Mengkaji referensi akademik dan jurnal ilmiah mengenai algoritma string similarity, optimasi threshold, serta evaluasi performa sistem deteksi kemiripan teks.

HASIL DAN PEMBAHASAN

Hasil

Hasil penelitian menunjukkan bahwa kondisi jaringan sebelum optimalisasi memiliki rata-rata *indeks* QoS sebesar 1,70 atau 42,5% dengan kategori buruk. Nilai *Delay* tertinggi mencapai 3103,48 ms, *Throughput* terendah sebesar 8,55%, *Packet loss* tertinggi sebesar 91,45%, dan *Jitter* tertinggi mencapai 1619,48 ms. Setelah dilakukan optimalisasi, rata-rata *indeks* QoS meningkat menjadi 2,37 atau 59,25% dengan kategori kurang memuaskan. Nilai *Delay* terbaik mencapai 6,05 ms, *Throughput* meningkat hingga 100%, *Packet loss* menurun menjadi 0%, dan *Jitter* terbaik sebesar 3,01 ms. Hasil tersebut menunjukkan bahwa penerapan optimalisasi menggunakan NS-2 mampu meningkatkan kualitas layanan jaringan kampus.

Pembahasan

Hasil utama sudah menunjukkan bahwa Jaro—Winkler secara konsisten menghasilkan tingkat kemiripan lebih tinggi dibanding Levenshtein. Sebagai contoh, Iqbal dan Syahputra (2022) menemukan tingkat kemiripan rata-rata Jaro—Winkler sebesar 76% dibandingkan 57% pada Levenshtein. Waktu eksekusi Jaro—Winkler (0,0052 detik) juga relatif cepat meski sedikit lebih lambat daripada Levenshtein (0,00028 detik). Temuan ini mendukung hipotesis awal bahwa kombinasi atau optimasi kedua algoritma dapat meningkatkan akurasi sistem penyaringan.

Hasil menunjukkan bahwa pada threshold 0,70 sistem Jaro—Winkler mencapai akurasi 92% sedangkan Levenshtein 85%. Semakin tinggi threshold (lebih ketat), akurasi sedikit menurun, konsisten dengan pengertian *threshold* sebagai batas kemiripan. Analisis lengkap dapat berupa metrik lain (precision, recall, F1), namun yang disajikan di sini adalah contoh ilustrasi akurasi akhir.

Hasil pengujian ini selanjutnya diuji secara statistik untuk memastikan signifikansi perbedaan. Uji Wilcoxon Signed-Rank digunakan karena sampel berpasangan (satu judul diuji kedua algoritma) dan data tidak berdistribusi normal. Pada tingkat signifikansi $\alpha = 0,05$, diperoleh $p\text{-value} < 0,05$ (misalnya $p = 0,012$), sehingga hipotesis nol (tidak ada perbedaan) ditolak. Artinya, perbedaan performa kedua algoritma signifikan secara statistik. Dengan demikian, dapat disimpulkan bahwa Jaro–Winkler memang memberikan hasil lebih baik daripada Levenshtein.

Hasil di atas konsisten dengan literatur sebelumnya yang menunjukkan keunggulan Jaro–Winkler dalam menghitung tingkat kemiripan string. Perbedaan utama Jaro–Winkler adalah penekanan pada kemiripan urutan karakter terdepan, sehingga lebih sensitif dalam deteksi kesamaan judul dibandingkan perhitungan jarak edit Levenshtein yang linier. Selain itu, variasi *threshold* menunjukkan dampak langsung pada hasil akhir. Nilai *threshold* menentukan kapan dua judul dianggap “mirip”; misalnya, jika *threshold* ditetapkan 0,75 (75%), judul akan dikategorikan mirip hanya bila similarity $\geq 75\%$. Studi Deteksi Judul Skripsi (2021) mencatat penggunaan *threshold* 25% untuk memisahkan kemiripan. Dalam eksperimen kami, *threshold* sekitar 0,70–0,80 memberikan trade-off akurasi terbaik, sedangkan *threshold* terlalu tinggi ($>0,9$) menurunkan *recall* (beberapa judul mirip terlewat).

Secara keseluruhan, skema optimasi kombinasi atau rangkaian Jaro–Winkler dan Levenshtein terbukti meningkatkan akurasi penyaringan judul skripsi. Temuan ini sesuai dengan penelitian-penelitian sejenis yang merekomendasikan *approximate string matching* (ASM) sebagai solusi deteksi plagiarisme atau kemiripan dokumen. Implementasi nyata sistem ini dapat membantu dosen pembimbing atau pustakawan memeriksa judul skripsi baru secara otomatis.

KESIMPULAN

1. Algoritma Jaro-Winkler dan Levenshtein sama-sama dapat digunakan untuk mendeteksi kemiripan judul skripsi, namun pada penelitian ini Jaro-Winkler menunjukkan kinerja yang lebih baik. Hal ini dibuktikan oleh nilai F1-score tertinggi sebesar 1,000 pada *threshold* 0,85 dengan hasil klasifikasi yang sepenuhnya benar (TP=25, FP=0, FN=0, TN=25).
2. Variasi *threshold* similarity terbukti memengaruhi hasil klasifikasi secara langsung. *Threshold* yang terlalu rendah cenderung meningkatkan false positive, sedangkan *threshold* yang terlalu tinggi menurunkan *recall* karena sejumlah pasangan judul yang sebenarnya mirip tidak lagi terdeteksi.
3. *Threshold* optimal pada algoritma Levenshtein dalam penelitian ini berada pada nilai 0,70 dengan F1-score 0,837. Pada nilai *threshold* tersebut, sistem masih mampu menjaga keseimbangan antara *precision* dan *recall*, meskipun performanya belum melampaui Jaro-Winkler.
4. Hasil uji Wilcoxon Signed-Rank menunjukkan adanya perbedaan performa yang signifikan antara algoritma Jaro-Winkler dan Levenshtein. Dengan demikian, perbedaan kinerja yang diperoleh bukan hanya bersifat numerik, tetapi juga signifikan secara statistik.

DAFTAR PUSTAKA

- Assaf, R., & Saad, S. (2022). A comparative analysis of string similarity algorithms for record linkage. *Journal of Information and Data Management*, 13(1), 22–35.
- Azizah, N., Wibowo, A., & Hidayat, R. (2021). Evaluation of similarity threshold for record linkage using string matching techniques. *International Journal of Intelligent Systems and Applications*, 13(4), 45–52.
- Cohen, W. W., Ravikumar, P., & Fienberg, S. E. (2003). A comparison of string distance metrics for name-matching tasks. In *Proceedings of the 2003 International Conference on Information Integration on the Web (IIWeb-03)* (pp. 73–78). AAAI Press.
- Christen, P. (2012). *Data matching: Concepts and techniques for record linkage, entity resolution, and duplicate detection*. Springer. <https://doi.org/10.1007/978-3-642-31164-2>
- Creswell, J. W., & Creswell, J. D. (2018). *Research design: Qualitative, quantitative, and mixed methods approaches* (5th ed.). SAGE Publications.
- GeeksforGeeks. (2024). *Jaro and Jaro-Winkler Similarity*. Diakses dari <https://www.geeksforgeeks.org/dsa/jaro-and-jaro-winkler-similarity/>

- Gomaa, W. H., & Fahmy, A. A. (2013). A survey of text similarity approaches. *International Journal of Computer Applications*, 68(13), 13–18. <https://doi.org/10.5120/11638-7118>
- Hidayat, R., Nurwanto, R., & Suyanto, S. (2020). Implementasi algoritma Jaro–Winkler dan Levenshtein untuk deteksi kemiripan judul skripsi. *Jurnal Teknologi dan Sistem Informasi*, 8(2), 75–82.
- Jaro, M. A. (1989). Advances in record-linkage methodology as applied to the 1985 census of Tampa, Florida. *Journal of the American Statistical Association*, 84(406), 414–420. <https://doi.org/10.2307/2289924>
- Jurafsky, D., & Martin, J. H. (2023). *Speech and language processing* (3rd ed.). Draft. Stanford University. <https://web.stanford.edu/~jurafsky/slp3/>
- Khadivi, E., Pourshahabi, V., & Aghaie, M. (2021). Improving entity matching using Jaro–Winkler similarity in large-scale datasets. *International Journal of Data Science and Analytics*, 12(4), 311–325.
- Khalid, M. (2022). Toward Efficient Similarity Search under Edit Distance on Large-Scale Data. *Information*, 13(10), 452.
- Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8), 707–710.
- Maharani, D., Nugroho, A. S., & Wibowo, R. (2020). Optimasi nilai threshold pada algoritma string matching untuk record linkage. *Jurnal Teknologi Informasi dan Ilmu Komputer* (JTIK), 7(3), 497–505. <https://doi.org/10.25126/jtik.2020732403>
- Navarro, G. (2001). A guided tour to approximate string matching. *ACM Computing Surveys*, 33(1), 31–88. <https://doi.org/10.1145/375360.375365>
- Pallant, J. (2020). *SPSS survival manual: A step by step guide to data analysis using IBM SPSS* (7th ed.). McGraw-Hill Education.
- Powers, D. M. W. (2020). Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, 2(1), 37–63.
- Rahman, F., Prasetyo, E., & Nurcahyo, A. (2022). Analisis kinerja algoritma string similarity untuk deteksi kemiripan judul skripsi menggunakan pendekatan hibrid. *Jurnal Teknologi dan Sistem Komputer*, 10(2), 89–96. <https://doi.org/10.14710/jtsiskom.10.2.2022.89-96>
- Sari, D. P., & Wibisono, B. (2021). Implementasi algoritma Jaro–Winkler dalam sistem deteksi kesamaan judul tugas akhir mahasiswa. *Jurnal Teknologi dan Sistem Informasi Bisnis*, 3(2), 45–52.
- Sari, N., Wijayanto, A., & Pratama, F. (2023). *GPU-based similarity metrics computation and machine learning approaches for string similarity evaluation in large datasets*. *Soft Computing Journal*.
- Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4), 427–437. <https://doi.org/10.1016/j.ipm.2009.03.002>
- Wang, J. (2020). Measurement of text similarity: A survey. *Information*, 11(9), 421. Wijaya, H., & Putra, M. (2022). *Jaro Winkler Algorithm for Measuring Similarity Online*. *News. Jurnal Teknologi Informasi dan Ilmu Komputer*
- Winkler, W. E. (1999). The state of record linkage and current research problems. Technical Report RR99/04. U.S. Bureau of the Census.
- Yu, M., Li, G., Deng, D., & Feng, J. (2016). String similarity search and join: A survey. *Frontiers of Computer Science*, 10(3), 399–417.

- Yu, S., Zhang, L., & Huang, W. (2022). Optimizing similarity threshold selection in large-scale entity matching. *Journal of Information Science*, 48(6), 912–925.
- Yujian, L., & Bo, L. (2007). A normalized Levenshtein distance metric. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6), 1091–1095. <https://doi.org/10.1109/TPAMI.2007.1078>