MENENTUKAN TINGKAT KEMIRIPAN JUDUL SKRIPSI MAHASISWA FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN UNISMUH MAKASSAR MENGGUKAN METODE *COSINE SIMILARITY*

SKRIPSI

Diajukan sebagai Salah Satu Syarat untuk Mendapatkan

Gelar Sarjana Komputer (S.Kom) Program Studi Informatika

HAEDIR

105841105620

PROGRAM STUDI INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS MUHAMMADIYAH MAKASSAR

2025

MENENTUKAN TINGKAT KEMIRIPAN JUDUL SKRIPSI MAHASISWA FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN UNISMUH MAKASSAR MENGGUKAN METODE *COSINE SIMILARITY*

Diajukan sebagai Salah Satu Syarat untuk Mendapatkan

Gelar Sarjana Komputer (S.Kom) Program Studi Informatika

Disusun Dan Diajukan Oleh:

HAEDIR

105841105620

PROGRAM STUDI INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS MUHAMMADIYAH MAKASSAR

2025

PENGESAHAN

Skripsi atas nama **Haedir** dengan nomor induk Mahasiswa **105 84 1105620**, dinyatakan diterima dan disahkan oleh Panitia Ujian Tugas Akhir/Skripsi sesuai dengan Surat Keputusan Dekan Fakultas Teknik Universitas Muhammadiyah Makassar Nomor 0004/SK-Y/55202/091004/2025 sebagai salah satu syarat guna memperoleh gelar Sarjana Komputer pada Program Studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar pada hari Sabtu, 30 Agustus 2025

Panitia Ujian :

Makassar, 06 Rabi'ul Awal 1447 H / 30 Agustus 2025 M

1. Pengawas Umum
   a. Rektor Universitas Muhammadiyah Makassar
      Dr. Ir. H. Abd. Rakhim Nanda, ST.,MT.,IPU : .....
   b. Dekan Fakultas Teknik Universitas Hasanuddin
      Prof. Dr. Eng. Muhammad Isran Ramli, S.T., M.T.(ASEAN.,Eng : .....
2. Penguji
   a. Ketua : Prof. Dr. Ir. Hafsah Nirwana, M.T. .....
   b. Sekertaris : Rizki Yusliana Bakti, S.T., M.T. .....
3. Anggota : 1. Muhyiddin A M Hayat, S.Kom, MT : .....
   2. Fahrim Irhamna Rachman, S.Kom., M.T. : .....
   3. Chyquitha Danuputri, S.Kom., M.Kom : .....

Mengetahui :

Pembimbing I	Pembimbing II
Lukman, S.Kom., M.T	**Titin Wahyuni, S.Pd., M.T.**

Dekan

Dr. Muh. Syafaat S. Kuba, S.T., M.T.
NBM : 795 288

HALAMAN PENGESAHAN

Tugas Akhir ini diajukan untuk memenuhi syarat ujian guna memperoleh gelar Sarjana Komputer (S.Kom) Program Studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar.

Judul Skripsi : MENENTUKAN TINGKAT KEMIRIPAN JUDUL SKRIPSI MAHASISWA FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN UNISMUH MAKASSAR MENGGUNAKAN METODE COSINE SIMILARITY

Nama : Haedir

Stambuk : 105 84 11056 20

Makassar, 10 September 2025

Telah Diperiksa dan Disetujui
Oleh Dosen Pembimbing;

Pembimbing I	Pembimbing II
Lukman, S.,Kom M.T.	**Titin Wahyuni, S.Pd., M.T.**

Mengetahui,
Ketua Prodi Informatika

Rizki Yusliana Bakti, S.T., M.T.
NBM : 1307 284

MOTTO DAN PERSEMBAHAN

Motto

"Jangan berhenti ketika lelah, berhentilah ketika selesai."

Persembahan

Dengan penuh rasa syukur ke hadirat Allah SWT, Skripsi ini saya persembahkan kepada kedua orang tua tercinta Bapak dan Ibu, kedua sosok yang tak pernah berhenti melimpahkan kasih sayang, doa tulus, serta dukungan yang tiada henti. Untuk Kedua Adik tercinta, dan keluarga besar yang selalu memberikan semangat, serta para dosen yang telah membimbing dan menuntun saya selama menempuh pendidikan. Ucapan terima kasih juga saya tunjukkan kepada sahabat-sahabat seperjuangan yang senantiasa hadir memberi warna dalam perjalanan ini. Untuk keluarga besar informatika 20, terima kasih atas semangat dan perjalanan yang dijalani bersama. terima kasih kepada seluruh pihak yang tidak dapat disebutkan satu per satu, atas segala dukungan dan doa yang tulus. Tak lupa, terima kasih kepada seluruh pihak yang tidak dapat disebutkan satu per satu, atas segala dukungan dan doa yang tulus. Dan akhir, untuk diriku sendiri, terima kasih telah bertahan, terus belajar, dan berjuang hingga titik ini, percaya bahwa setiap niat baik dan harapan akan selalu diberikan kemudahan oleh-Nya. Semoga persembahan ini bermanfaat bagi pengembangan ilmu pengetahuan dan menjadi bagian dari amal kebaikan. Selamat bergelar S.Kom.

ABSTRAK

HAEDIR . Menentukan tingkat kemiripan judul skripsi mahasiswa fakultas keguruan dan ilmu pendidikan unismuh makassar menggunakan metode *cosine similarity* (dibimbing oleh Lukman, S.Kom., M.T. dan Titin Wahyuni,S.Pd., M.T.)

Plagiarisme dalam dunia akademik merupakan permasalahan yang serius, khususnya dalam penentuan judul skripsi yang orisinal dan unik. Penelitian ini bertujuan untuk mengimplementasikan metode *Cosine Similarity* dalam mendeteksi tingkat kemiripan antar judul skripsi mahasiswa Fakultas Keguruan dan Ilmu Pendidikan (FKIP) Universitas Muhammadiyah Makassar. Metode ini bekerja dengan mengukur sudut kemiripan antara dua vektor teks yang telah direpresentasikan menggunakan teknik *Term Frequency-Inverse Document Frequency* (TF-IDF). Penelitian dilakukan melalui tahapan pengumpulan data, preprocessing (tokenisasi, *stopword* removal, dan *stemming*), ekstraksi fitur TF-IDF, perhitungan nilai *cosine similarity*, serta evaluasi performa menggunakan metrik akurasi, presisi, *recall*, dan *F1-score*. Sistem yang dikembangkan mampu mengklasifikasikan judul sebagai "mirip" atau "tidak mirip" berdasarkan ambang batas (threshold) yang ditentukan. Hasil evaluasi menunjukkan akurasi sebesar 87,33%, presisi 100%, *recall* 58,70%, dan *F1-score* 73,97%. Temuan ini mengindikasikan bahwa metode *Cosine Similarity* efektif dalam mendeteksi kemiripan judul, meskipun masih terdapat ruang untuk peningkatan terutama dalam hal sensitivitas. Penelitian ini diharapkan dapat menjadi referensi bagi institusi akademik dalam menjaga orisinalitas karya ilmiah mahasiswa.

Kata Kunci: *Cosine Similarity*, TF-IDF, Kemiripan Judul, Deteksi Plagiarisme, Natural Language Processing

ABSTRACT

HAEDIR . Menentukan tingkat kemiripan judul skripsi mahasiswa fakultas keguruan dan ilmu pendidikan unismuh makassar menggunakan metode *cosine similarity* (dibimbing oleh Lukman, S.Kom., M.T. dan Titin Wahyuni,S.Pd., M.T.)

Plagiarism in academia is a serious issue, especially in ensuring the originality and uniqueness of undergraduate thesis titles. This study aims to implement the Cosine Similarity method to detect the degree of similarity between thesis titles of students from the Faculty of Teacher Training and Education (FKIP) at Universitas Muhammadiyah Makassar. The method works by calculating the cosine of the angle between two text vectors represented using Term Frequency-Inverse Document Frequency (TF-IDF). The research involved several stages, including data collection, preprocessing (tokenization, stop word removal, and stemming), TF-IDF feature extraction, similarity computation, and performance evaluation using metrics such as accuracy, precision, recall, and F1-score. The developed system classifies titles as "similar" or "not similar" based on a predefined threshold. Evaluation results show an accuracy of 87.33%, precision of 100%, recall of 58.70%, and F1-score of 73.97%. These findings indicate that the Cosine Similarity method is effective in identifying title similarities, although there is room for improvement in sensitivity. This study is expected to serve as a reference for academic institutions in promoting originality in student research works.

***Keywords**: Cosine Similarity, TF-IDF, Title Similarity, Plagiarism Detection, Natural Language Processing*

KATA PENGANTAR

Alhamdulillah, puji dan syukur penulis panjatkan kepada Allah SWT, Tuhan semesta alam, Dzat yang selalu melimpahkan rahmat dan hidayah- Nya, sehingga penulis dapat menyelesaikan penelitian tugas akhir yang berjudul "Menentukan Tingkat Kemiripan Judul Mahasiswa Fakultas Keguruan Dan Ilmu Pendidikan Unismuh Makassar Menggunakan Metode Cosine Similarity (Implementasi dan Penilaian Kinerja Metode)". Tak lupa selawat dan salam senantiasa tercurah atas junjungan kita Nabi Muhammad SAW, Kedua Orang Tua Yang Tercinta, keluarga, sahabat, dan umatnya hingga akhir zaman. Aamiin. Tugas akhir ini disusun sebagai salah satu persyaratan yang harus dipenuhi dalam menyelesaikan jenjang Strata-1 pada program studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar.

Selama proses penyusunan skripsi ini, penulis menerima banyak bimbingan, arahan, motivasi, dan bantuan dari berbagai pihak, baik secara langsung maupun tidak langsung. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan terima kasih kepada:

1. Bapak Dr. Ir. H. Abd Rakhim Nanda, S.T., M.T IPU. sebagai Rektor Universitas Muhammadiyah Makassar.
2. Bapak Ir. Muhammad Syafa'at S.Kuba. ST., M.T Sebagai Dekan Fakultas Teknik Universitas Muhammadiyah Makassar.
3. Bapak Muhyiddin A M Hayat, S.Kom., M.T sebagai Ketua Prodi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar.
4. Bapak Lukman, S.Kom., M.T sebagai pembimbing I yang dengan telah Ikhlas memberikan bimbingan dan arahan selama penyusunan tugas akhir ini.
5. Ibu Titin Wahyuni, S.Pd., M.T sebagai pembimbing II yang dengan telah Ikhlas memberikan bimbingan dan arahan selama penyusunan vi tugas akhir ini. V
6. Segenap Bapak – bapak dan Ibu Dosen Prodi Informatika Fakultas Teknik

Universitas Muhammadiyah Makassar yang telah memberikan bakat dan ilmu pengetahuan serta mendidik penulis selama proses belajar mengajar di Universitas Muhammadiyah Makassar.

7. Rekan-rekan mahasiswa utamanya angkatan 2020 Fakultas Teknik Universitas Muhammadiyah Makassar terima kasih atas dukungan dan kerjasamanya selama menempuh Pendidikan serta penyelesaian penyususan proposal skripsi ini.

8. Terakhir, terima kasih untuk diri sendiri, karena telah mampu berusaha keras dan berjuang sampai sejauh ini.

Pada fase dewasa ini, kadangkala kita lupa akan arti kebaikan, kebenaran, kesetiaan, persahabatan, ketenangan, dan cinta, karena terlalu banyak penderitaan yang kita peroleh. Tetapi suatu hal yang harus diingat "Bukan rasa sulitlah yang membuat kita takut, tapi rasa takutlah yang membuat kita sulit"

Skripsi ini saya persembahkan untuk orang-orang yang selalu bertanya "kapan skripsimu selesesai?" dan "kapan kamu wisuda". Terlambat lulus atau lulus tidak tepat waktu bukanlah sebuah kejahatan, bukan pula sebuah aib. Alangkah kerdilnya jika mengukur kecerdasan seseorang hanya dari siapa yang paling cepat lulus. Bukankah sebaik baiknya skripsi adalah skripsi yang selesai? Mungkin ada suatu hal dibalik terlambatnya mereka lulus, dan percayalah, alasan saya disini adalah alasan yang sepenuhnya baik.

Makassar, 2 Agustus 2024

Haedir

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR ISTILAH

Cosine Si`milarity	Metode untuk mengukur kesamaan antara dua vektor dalam ruang multidimensi dengan menghitung cosinus sudut di antara keduanya.
TF-IDF (Term Frequency-Inverse Document Frequency)	Metode untuk memberikan bobot pada kata-kata dalam dokumen berdasarkan frekuensi kemunculannya dan seberapa unik kata tersebut di seluruh dokumen.
Python	Bahasa pemrograman tingkat tinggi yang sering digunakan dalam pengolahan data, pembelajaran mesin, dan pengembangan aplikasi.
Visual Studio Code	Editor kode sumber yang ringan namun kaya fitur, digunakan untuk pengembangan perangkat lunak termasuk pemrograman Python.
Stop Words Removal	Proses menghapus kata-kata umum (seperti "dan", "atau", "yang") dari teks untuk meningkatkan efisiensi analisis teks.
Stemming	Teknik untuk mengurangi kata ke bentuk dasarnya dengan menghilangkan imbuhan, seperti "berlari" menjadi "lari".

Accuracy	Ukuran seberapa banyak judul yang diklasifikasikan (mirip atau tidak mirip) dengan benar oleh metode, dibandingkan dengan total jumlah judul.
Precision	Dari semua judul yang diprediksi mirip oleh metode, berapa banyak yang sebenarnya memang mirip.
Recall	Metrik evaluasi yang mengukur kemampuan model dalam menemukan semua instance yang relevan.
F1-Score	Ukuran kinerja model yang menggabungkan presisi dan recall dalam satu metrik harmonis.
Tokenisasi	Proses memecah teks menjadi unit-unit kecil seperti kata, frasa, atau kalimat untuk analisis lebih lanjut.

BAB I
PENDAHULUAN

A. Latar Belakang

Dalam era digital yang serba cepat seperti sekarang, teknologi informasi telah menjadi bagian penting dalam dunia akademik. Fakultas Keguruan dan Ilmu Pendidikan (FKIP) di Universitas Muhammadiyah (Unismuh) Makassar menghadapi tantangan dalam mengelola dan menilai kesamaan judul skripsi mahasiswa. Keunikan dan orisinalitas judul skripsi sangatlah penting untuk mencegah plagiarisme dan memastikan keragaman topik penelitian.

Seiring berkembangnya teknologi kecerdasan buatan, terutama di bidang pengolahan bahasa alami atau Natural Language Processing (NLP), berbagai metode analisis teks mulai diterapkan. NLP memungkinkan komputer memahami dan mengolah bahasa manusia, yang menjadi dasar untuk analisis kemiripan teks.

Plagiarisme merupakan salah satu masalah serius dalam dunia akademik. Untuk mengatasi masalah ini, diperlukan penerapan dan pemahaman mendalam terhadap metode yang dapat menganalisis kemiripan teks, guna mengarahkan pada penggunaan yang lebih baik. Metode cosine similarity adalah salah satu pendekatan yang umum digunakan untuk menghitung tingkat kesamaan antara dua item teks, seperti judul skripsi. Pendekatan ini bekerja dengan cara membandingkan representasi numerik dari item-item tersebut. Dengan adanya pembobotan TF-IDF (Term Frequency - Inverse Document Frequency), metode cosine similarity dapat memproses kata secara lebih optimal (Apriani, Hizbu, Khairan, 2021).

Penelitian ini bertujuan untuk mengimplementasikan metode cosine similarity dan mengetahui kinerjanya dalam mendeteksi kemiripan judul skripsi mahasiswa di FKIP Unismuh Makassar. Fokus utama adalah pada pemahaman sejauh mana metode ini mampu memberikan hasil yang akurat dan relevan

dalam konteks dataset yang ada, sehingga dapat menjadi panduan untuk penerapannya yang lebih baik. Penelitian ini akan melibatkan tahapan implementasi metode, pengujian dengan data riil, dan evaluasi hasil menggunakan metrik standar seperti akurasi, presisi, recall, dan F1-score.

Penggunaan metode cosine similarity untuk penelitian ini didasarkan pada relevansinya dalam studi kemiripan teks. Penelitian ini juga akan mempertimbangkan keterbatasan cosine similarity dalam mengenali sinonim (Sutikno & Saniati, 2021) sebagai bagian dari diskusi hasil evaluasi.

Dengan mengimplementasikan dan mengetahui kinerja metode cosine similarity, diharapkan penelitian ini dapat memberikan pemahaman konkret mengenai kemampuannya dalam membantu proses verifikasi keunikan judul skripsi di Universitas Muhammadiyah (Unismuh) Makassar. Penelitian ini bukan merupakan pengembangan sistem machine learning yang melibatkan pelatihan model prediktif kompleks, melainkan sebuah studi implementatif dan evaluatif terhadap sebuah metode pengukuran kemiripan teks.

B. Rumusan Masalah

1. Bagaimana cara mengimplementasikan metode cosine similarity untuk mendeteksi kesamaan judul skripsi mahasiswa di FKIP Unismuh Makassar?
2. Bagaimana cara mengetahui kinerja metode cosine similarity dalam mendeteksi kesamaan judul skripsi mahasiswa di FKIP Unismuh Makassar berdasarkan hasil evaluasi (akurasi, presisi, recall, F1-score), guna mengarahkan pada penerapannya yang lebih optimal?

C. Tujuan Penelitian

1. Mengimplementasikan metode cosine similarity untuk mendeteksi kesamaan judul skripsi mahasiswa FKIP Unismuh Makassar.
2. Mengetahui dan mengevaluasi kinerja metode cosine similarity dalam mendeteksi kesamaan judul skripsi mahasiswa FKIP Unismuh Makassar

menggunakan metrik akurasi, presisi, recall, dan F1-score, sebagai dasar untuk penggunaan yang lebih baik.

D. Manfaat Penelitian

1. Bagi Fakultas dan Program Studi

   Memberikan pemahaman empiris mengenai kinerja metode cosine similarity yang dapat menjadi dasar pertimbangan dan panduan untuk penerapannya secara lebih baik dalam membantu proses verifikasi keunikan judul skripsi.

2. Bagi Mahasiswa

   Meningkatkan kesadaran akan pentingnya orisinalitas judul dan memberikan pemahaman tentang bagaimana teknologi dapat digunakan untuk mendeteksi kemiripan.

3. Bagi Pengembangan Ilmu

   Menambah studi kasus implementasi dan evaluasi metode cosine similarity dalam konteks data akademik di Indonesia, khususnya di lingkungan Unismuh Makassar, serta memberikan wawasan untuk pengembangan metode yang lebih lanjut.

E. Ruang Lingkup Penelitian

Penelitian ini memiliki ruang lingkup sebagai berikut:

1. Fokus Metode

   Implementasi metode *cosine similarity* untuk mengukur kemiripan antar judul skripsi.

2. Objek Data

   Judul-judul skripsi mahasiswa dari Fakultas Keguruan dan Ilmu Pendidikan (FKIP) Universitas Muhammadiyah Makassar.

3. Tahapan Penelitian:

   a. Pengumpulan data judul skripsi.

   b. *Preprocessing* data (normalisasi, tokenisasi, stopword removal, dll).

c. Perhitungan bobot kata menggunakan metode TF-IDF.
d. Pengukuran kemiripan antar judul menggunakan *cosine similarity*.
e. Evaluasi hasil menggunakan metrik evaluasi yang telah ditentukan.

4. Bentuk aplikasi

Benruk penerapan *cosine similarity* dalam menentukan Tingkat kemiripan judul adalah aplikasi berbasis web

F. Sistematika Penulisan

Secara garis besar penulisan laporan tugas akhir ini terbagi menjadi beberapa bab yang tersusun yaitu:

BAB I PENDAHULUAN

Bab ini menjelaskan tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan. Tujuan

BAB II TINJAUAN PUSTAKA

Bab ini menjelaskan tentang teori-teori yang melandasi penulisan dalam melaksanakan skripsi.

BAB III METODE PENELITIAN

Membahas tentang metode penelitian dan alat yang digunakan untuk pembuatan sistem.

BAB IV HASIL PENELITIAN

Pada bab ini menjelaskan hasil dari penelitian yang sudah di lakukan sebelumnya, pada bab inilah di jelaskan hasil penelitian dan pengujian.

BAB V PENUTUP

Pada bab ini dijelaskan kesimpulan yang di hasilkan dari penelitian yang telah di lakukan dan saran yang di berikan kepada penelti selanjutnya

BAB II
TINJAUAN PUSTAKA

A. Landasan Teori

1. Teknologi Informasi dalam Dunia Akademik

Teknologi informasi telah menjadi bagian integral dalam dunia akademik, khususnya dalam pengelolaan dan penilaian kesamaan judul skripsi mahasiswa. Kehadiran teknologi informasi memberikan kemudahan dalam memproses, menganalisis, dan mengelola data akademik secara efisien. Dalam konteks ini, sistem digital memungkinkan institusi pendidikan untuk melakukan deteksi kesamaan atau kemiripan judul secara cepat dan akurat, sehingga dapat meminimalkan praktik plagiarisme serta menjaga orisinalitas karya ilmiah mahasiswa.

Teknologi informasi mencakup segala proses yang berkaitan dengan penggunaan alat bantu dalam manipulasi, pengelolaan, dan perpindahan informasi antar media. Perpaduan antara teknologi dan informasi tidak dapat dipisahkan karena keduanya memiliki keterkaitan erat dalam menciptakan sistem yang akurat, teratur, akuntabel, dan terpercaya. Dalam dunia pendidikan, teknologi informasi menjadi komponen penting yang mendukung pembelajaran dan tata kelola akademik yang lebih modern dan efisien (Zahwa & Syafi'i, 2022).

2. Natural Language Processing (NLP)

Natural Language Processing (NLP) adalah bidang ilmu yang berkaitan dengan interaksi antara komputer dan bahasa alami manusia. Dalam konteks penelitian ini, NLP berperan penting dalam tahap pra-pemrosesan teks, seperti pembersihan, tokenisasi, dan transformasi data teks menjadi bentuk yang dapat dianalisis secara matematis. Proses ini menjadi dasar utama sebelum data dianalisis menggunakan metode cosine similarity untuk mengukur tingkat kemiripan antar teks.

NLP merupakan subbidang dari kecerdasan buatan yang bertujuan untuk memahami, memproses, dan menghasilkan informasi berbasis teks dengan cara yang meniru kemampuan manusia dalam berbahasa. Dengan memanfaatkan NLP, sistem komputer dapat mengidentifikasi makna, struktur, dan konteks dalam teks secara lebih mendalam, sehingga memungkinkan analisis semantik yang lebih akurat (Nurwanda *et al.*, 2024).

3. Plagiarisme dalam Dunia Akademik

Plagiarisme merupakan permasalahan serius yang terus menjadi perhatian dalam dunia akademik. Tindakan ini tidak hanya merusak integritas ilmiah, tetapi juga menghambat perkembangan pengetahuan yang orisinal. Oleh karena itu, penerapan metode deteksi plagiarisme yang efektif sangat penting untuk memastikan bahwa karya ilmiah yang dihasilkan benar-benar mencerminkan pemikiran dan usaha mandiri dari penulisnya.

Menurut Manullang et al. (2021), plagiarisme dapat diartikan sebagai tindakan mengambil, menerbitkan, atau mengklaim hasil pemikiran orang lain sebagai milik sendiri. Dalam konteks akademik, praktik ini termasuk dalam pelanggaran etik yang dapat dikenakan sanksi. Untuk mencegah hal tersebut, berbagai sistem deteksi seperti Turnitin telah banyak digunakan sebagai alat bantu dalam memverifikasi orisinalitas karya ilmiah.

4. Cosine similarity

Cosine similarity adalah metode yang digunakan untuk mengukur tingkat kesamaan antara dua teks berdasarkan nilai cosinus dari sudut antara dua vektor dalam ruang multidimensi. Setiap dokumen atau teks diubah terlebih dahulu ke dalam representasi numerik melalui model ruang vektor, biasanya menggunakan teknik seperti TF-IDF. Metode ini sangat relevan dalam mengevaluasi tingkat kemiripan judul skripsi karena mampu menangkap hubungan semantik antar kata secara kuantitatif.

Menurut Anugrah (2021), cosine similarity membandingkan vektor query dengan vektor dokumen untuk menentukan seberapa besar kemiripan antara keduanya. Semakin kecil sudut antara dua vektor, semakin tinggi tingkat kesamaan kontennya. Dalam penelitian ini, metode cosine similarity akan diimplementasikan dan diuji untuk mengukur efektivitasnya dalam mendeteksi kemiripan antar judul skripsi mahasiswa secara sistematis dan terukur.

5. Proses Preprocessing Data

Proses preprocessing data meliputi tokenisasi, penghapusan stop words, dan stemming.

- **Tokenisasi**: Memecah teks menjadi unit-unit kecil.
- **Stop Words Removal**: Menghapus kata-kata umum (misalnya, "dan", "atau", "yang"). Meskipun tujuannya adalah menyeleksi makna, penghapusan stop words membantu mengurangi noise dan dimensi data, sehingga perhitungan kemiripan dapat lebih fokus pada kata-kata yang memiliki bobot informatif lebih tinggi.
- Stemming: Mengurangi kata-kata ke bentuk dasarnya (misalnya, "menganalisis", "analisis" menjadi "analis"). Ini membantu mengkonsolidasikan kata-kata yang memiliki akar makna yang sama, sehingga variasi bentuk kata tidak dianggap sebagai entitas yang sama sekali berbeda, yang penting untuk menangkap kesamaan makna inti meskipun ada perbedaan gramatikal.

Tahapan ini krusial untuk menghasilkan representasi numerik yang lebih bersih dan konsisten untuk analisis cosine similarity, yang pada gilirannya mempengaruhi akurasi perhitungan kemiripan.

6. Evaluasi untuk Menilai Kinerja Metode Deteksi Kemiripan

Untuk menilai kinerja suatu metode deteksi kemiripan, diperlukan metrik evaluasi yang standar. Beberapa metrik yang umum digunakan adalah:

- **Akurasi (Accuracy)**: Mengukur proporsi prediksi yang benar (baik mirip maupun tidak mirip) dari keseluruhan data.
- **Presisi (Precision)**: Mengukur proporsi judul yang diprediksi mirip yang sebenarnya memang mirip. Penting untuk meminimalkan false positive.
- **Recall (Sensitivity)**: Mengukur proporsi judul yang sebenarnya mirip yang berhasil dideteksi oleh metode. Penting untuk meminimalkan false negative.

F1-Score: Merupakan rata-rata harmonik dari presisi dan recall, memberikan keseimbangan antara keduanya. Nilai F1-Score yang tinggi menunjukkan bahwa metode memiliki presisi dan recall yang baik, yang mengindikasikan kinerja yang baik.

7. Web

Web Server merupakan sebuah perangkat lunak dalam server yang berfungsi menerima permintaan (request) berupa halaman web melalui HTTP atau HTTPS dari klien yang dikenal dengan browser web dan mengirimkan kembali (response) hasilnya dalam bentuk halaman-halaman web yang umumnya berbentuk dokumen HTML. Web server adalah perangkat lunak yang berfungsi sebagai penerima permintaan yang dikirimkan melalui browser kemudian memberikan tanggapan permintaan dalam bentuk halaman situs web atau lebih umumnya dalam dokumen HTML

B. Penelitian Terkait

Penelitian-penelitian sebelumnya menunjukkan berbagai pendekatan dalam pemanfaatan teknologi untuk meningkatkan akurasi dalam pencarian informasi akademik dan deteksi plagiarisme.

Penelitian terbaru oleh Aidhil Prima Abdi Guna (2024) mengangkat penerapan algoritma Cosine Similarity dalam konteks yang berbeda, yaitu untuk meningkatkan efektivitas pengacakan soal ujian online. Penelitian ini menggunakan pendekatan TF-IDF untuk merepresentasikan soal ke dalam bentuk vektor numerik, sebelum dilakukan perhitungan nilai kemiripan menggunakan algoritma Cosine Similarity. Untuk mengevaluasi performanya, digunakan metrik Mean Absolute Error (MAE) dan Root Mean Squared Error (RMSE). Hasil yang diperoleh menunjukkan bahwa selisih antara distribusi aktual dan prediksi sangat kecil, yang berarti sistem berhasil melakukan pengacakan soal secara adil dan efisien. Meskipun konteks penggunaannya bukan untuk pencocokan judul, penelitian ini menunjukkan bahwa algoritma Cosine Similarity memiliki potensi tinggi dalam menangani persoalan yang melibatkan pembandingan teks berbasis kemiripan.

Penelitian mengenai deteksi kemiripan judul skripsi semakin banyak dilakukan seiring meningkatnya kebutuhan akan sistem validasi judul yang efisien dan akurat dalam lingkungan akademik. Salah satu penelitian yang menonjol adalah yang dilakukan oleh Lindang, Jumaidil, dan Aswin (2022), yang merancang dan menerapkan sistem penentuan kemiripan antar judul skripsi menggunakan metode *Cosine Similarity*. Sistem ini dibangun dengan teknologi pemrograman PHP dan basis data MySQL. Penelitian ini dilakukan dengan tujuan untuk membantu mahasiswa dalam menemukan referensi skripsi yang memiliki kesamaan topik atau tema, serta mencegah plagiarisme secara tidak langsung. Berdasarkan pengujian yang dilakukan, sistem yang dikembangkan berhasil mencapai rata-rata akurasi sebesar 97%, dengan rata-rata waktu pemrosesan sebesar 0,277154 detik. Hal ini menunjukkan bahwa

penggunaan metode *Cosine Similarity* dalam konteks ini tidak hanya efektif dalam mengukur tingkat kemiripan, tetapi juga efisien dari segi waktu proses komputasi. Sistem ini juga memungkinkan pencarian cepat dan akurat terhadap skripsi yang memiliki topik serupa berdasarkan kata kunci yang dimasukkan pengguna.

Selanjutnya, penelitian oleh Pernanda dan Hakiki (2021) juga mengangkat permasalahan serupa, yakni pendeteksian kemiripan judul skripsi mahasiswa. Penelitian ini diterapkan di lingkungan STKIP PGRI Sumatera Barat, dengan fokus utama pada pengembangan sistem yang dapat memudahkan validator dalam proses persetujuan judul yang diajukan oleh mahasiswa. Penelitian ini menekankan pentingnya keberadaan sistem yang mampu mengurangi potensi pengajuan judul skripsi yang serupa atau bahkan identik. Dengan menggunakan metode *Cosine Similarity*, sistem yang dikembangkan mampu melakukan pencocokan terhadap database judul skripsi yang telah tersimpan sebelumnya. Melalui sistem ini, validator dapat melihat daftar judul dengan tingkat kemiripan tertentu, sehingga keputusan terhadap kelayakan judul dapat dilakukan secara lebih objektif dan berbasis data. Meskipun tidak disebutkan angka akurasi secara eksplisit, hasil dari sistem ini dinilai sangat membantu dalam proses seleksi dan validasi judul secara lebih cepat dan terstruktur.

Fitrianingsih et al. (2022) mengembangkan sebuah sistem untuk mendeteksi kemiripan judul skripsi secara otomatis guna menghindari duplikasi judul di lingkungan akademik. Penelitian ini menggunakan algoritma Oliver, yang diimplementasikan dalam bahasa pemrograman PHP untuk mengukur kemiripan antar string judul. Dengan menggunakan 10 judul uji dan 217 judul sebagai data training, algoritma ini berhasil mendeteksi dua judul yang memiliki tingkat kemiripan tinggi dan tidak layak untuk digunakan karena melebihi ambang batas kemiripan. Hasil penelitian ini menunjukkan

pentingnya penggunaan sistem pendeteksi kemiripan sebagai kontrol awal dalam pengajuan judul skripsi.

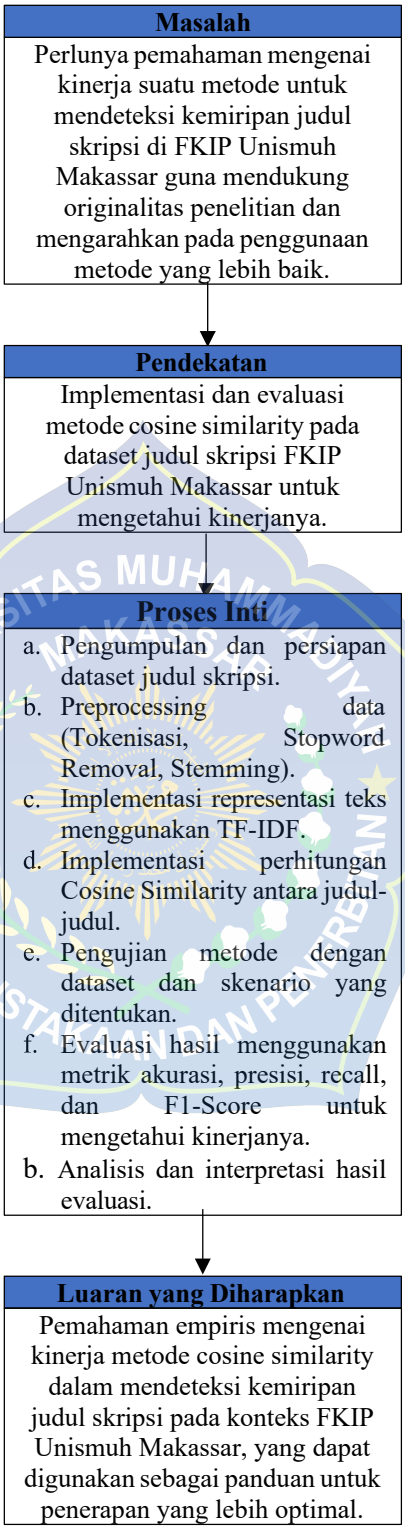
Penelitian yang dilakukan oleh Haruna, Bakti, dan Wahyuni (2024) mengkaji proses deteksi tingkat kemiripan judul skripsi pada Program Studi Informatika dengan memanfaatkan kombinasi metode *Support Vector Machine* (SVM) dan *Natural Language Processing* (NLP). Latar belakang penelitian ini berangkat dari pentingnya menghindari kemiripan judul untuk mencegah plagiarisme akademik. Dalam studinya, peneliti mengembangkan sistem yang mampu mengidentifikasi judul skripsi yang memiliki tingkat kesamaan tinggi dengan data judul yang sudah ada. Proses yang dilakukan meliputi pengumpulan data judul dari basis data, kemudian dilakukan *preprocessing* seperti tokenisasi, penghapusan stopword, dan stemming. Data yang telah dibersihkan kemudian diubah menjadi vektor numerik menggunakan metode *Term Frequency–Inverse Document Frequency* (TF-IDF). Hasil representasi ini digunakan sebagai input ke dalam model SVM untuk melakukan klasifikasi kemiripan. Model yang dikembangkan mampu mendeteksi judul-judul serupa dengan akurasi yang cukup tinggi, menunjukkan bahwa pendekatan kombinasi SVM dan NLP efektif dalam mengatasi masalah plagiarisme judul skripsi. Penelitian ini menjadi referensi penting dalam pengembangan sistem deteksi otomatis, karena membuktikan bahwa metode berbasis machine learning mampu menangani variasi bahasa dan struktur kalimat dalam judul secara efisien.

Penelitian lebih lanjut dilakukan oleh Nasrullah (2024) yang mengintegrasikan metode pembobotan *TF-IDF* dengan algoritma *Cosine Similarity* dalam mendeteksi tingkat kemiripan judul tugas akhir mahasiswa di Universitas Ichsan Gorontalo. Penelitian ini mengadopsi pendekatan *text preprocessing* secara lengkap, mencakup *case folding*, *tokenizing*, *stopword removal*, dan *stemming*, untuk memastikan bahwa data teks yang dianalisis berada dalam bentuk yang optimal. Setelah dilakukan pengolahan data dan

pengujian terhadap beberapa skenario, sistem yang dikembangkan menunjukkan performa yang cukup baik dengan hasil rata-rata akurasi sebesar 89,7%, precision 72,4%, dan recall sebesar 94,6%. Angka-angka ini menunjukkan bahwa sistem memiliki kemampuan tinggi dalam mengidentifikasi judul-judul yang mirip secara substansial. Penelitian ini juga memberikan kontribusi dalam hal penerapan teknologi pengolahan bahasa alami (*Natural Language Processing*) pada bidang akademik, khususnya dalam deteksi kesamaan konten teks ilmiah.

Adapun penelitian oleh Nugroho, Ramadhan, dan Khusaini (2021) mengambil pendekatan yang lebih luas, yaitu pada pengembangan sistem informasi manajemen proposal penelitian dan pengabdian kepada masyarakat (PKM) di lingkungan Universitas Pamulang. Dalam sistem ini, diterapkan algoritma *Cosine Similarity* untuk mendeteksi kemiripan isi proposal antara satu pengajuan dengan lainnya. Tujuan utama dari sistem ini adalah untuk mencegah pengajuan proposal yang terlalu mirip, sekaligus memudahkan reviewer atau pengelola kegiatan PKM dalam memfilter dan menilai substansi proposal secara lebih akurat. Dengan mengelola lebih dari 2.600 data dosen, sistem ini terbukti mampu meningkatkan efisiensi penyimpanan data dan mempercepat proses pencarian serta seleksi proposal yang memiliki topik atau tujuan serupa. Penggunaan metode *Cosine Similarity* dalam konteks ini juga dinilai berhasil dalam mendukung transparansi dan kualitas penelitian yang diajukan.

C. Kerangka Berpikir



BAB III
METODE PENELITIAN

A. Tempat dan Waktu Penelitian

Tempat Penelitian adalah suatu tempat atau objek yang akan dilakukan suatu penelitian. Penentuan lokasi penelitian merupakan langkah penting dalam proses penelitian karena memudahkan peneliti untuk melakukan penelitian. Lokasi penelitian yang dipilih oleh penulis adalah di Universitas Muhammadiyah Makassar. Tempat atau wilayah tersebut dipilih oleh penulis dengan alasan karena tempat penelitian atau lebih tepatnya Fakultas Keguruan Dan Ilmu Pendidikan Universitas Muhammadiyah Makassar pada proses pembuatan jurnal ilmiah masih dilakukan dengan cara manual sehingga seringkali menguras banyak waktu dan tenaga mahasiswa, oleh karena itu peneliti memilih tempat penelitian tersebut dikarenakan masalah yang diangkat oleh peneliti sehaluan dengan tempat tersebut.

Waktu penelitian ini akan dilakukan dalam jangka waktu kurang lebih 2 bulan, yaitu dimulai pada bulan Januari 2025 sampai dengan Maret 2025.

B. Alat Dan Bahan

Untuk melaksanakan penelitian diperlukan alat dan bahan yang akan digunakan pada penelitian ini yaitu :

1. Laptop
2. Data Skripsi
3. *Python*
4. *Visual Studio Code*

C. Perancangan Sistem

Penelitian ini dirancang mengikuti tahapan implementasi dan evaluasi kemiripan judul skripsi menggunakan pendekatan Cosine Similarity terhadap representasi vektor TF-IDF, dengan langkah-langkah sebagai berikut:

1. Pengumpulan

Langkah pertama adalah mengumpulkan data berupa judul-judul skripsi dari mahasiswa FKIP Universitas Muhammadiyah Makassar. Sebagian data

digunakan sebagai data latih, dan sebagian lainnya disiapkan sebagai data uji (test set) yang akan divalidasi menggunakan label manual atau anotasi dari dosen sebagai ground truth.

2. Preprocessing Data

   Data judul yang telah dikumpulkan kemudian diproses secara teks dengan beberapa tahap:

   a. Tokenisasi, yaitu memecah teks judul menjadi unit kata.
   b. Stop Words Removal, untuk menghapus kata-kata umum yang tidak relevan (seperti "yang", "dalam", "dari").
   c. Stemming, yaitu mengembalikan kata ke bentuk dasar agar kata-kata turunan dihitung sebagai satu entitas yang sama.

3. Ekstraksi Fitur

   Setelah preprocessing, setiap judul dikonversi menjadi representasi numerik menggunakan teknik TF-IDF (Term Frequency-Inverse Document Frequency). Bobot TF-IDF menunjukkan tingkat kepentingan suatu kata dalam sebuah judul relatif terhadap keseluruhan koleksi dokumen.

4. Cosine

   Langkah ini menghitung skor cosine similarity antara dua judul skripsi berdasarkan vektor TF-IDF mereka. Cosine similarity mengukur derajat kesamaan berdasarkan sudut antar vektor, dengan nilai 1 menunjukkan kemiripan sempurna dan 0 menunjukkan tidak ada kemiripan.

5. Evaluasi Model

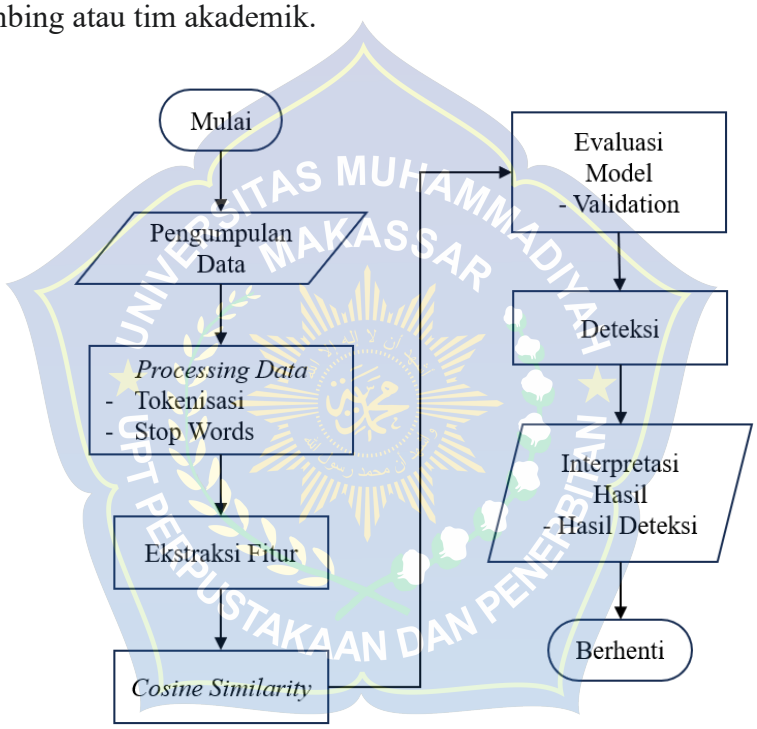
   Model dievaluasi menggunakan data uji. Hasil cosine similarity dibandingkan dengan label kemiripan manual untuk mengukur akurasi sistem. Validasi ini penting untuk mengetahui apakah sistem mendeteksi kemiripan sebagaimana yang dinilai oleh pakar.

6. Deteksi

Berdasarkan nilai cosine similarity dan ambang batas (threshold) yang telah ditentukan, sistem akan mengklasifikasikan setiap pasangan judul sebagai "mirip" atau "tidak mirip".

7. Interpretasi Hasil

Tahap akhir adalah menyajikan hasil deteksi dalam bentuk output yang mudah dipahami. Sistem akan menampilkan skor kemiripan dan status deteksi (mirip/tidak mirip) sebagai dasar pengambilan keputusan lebih lanjut oleh dosen pembimbing atau tim akademik.



Gambar 1. Flowchart Sistem

D. Teknik Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa sistem yang dibangun sudah sesuai dengan analisis dan tujuan penelitian. Berikut adalah langkah-langkah pengujian sistem:

1. Persiapan Data Uji (Ground Truth):
   - Sebuah subset dari dataset judul akan dipilih sebagai data uji.
   - Pasangan-pasangan judul dalam data uji ini akan dinilai secara manual oleh ahli (misalnya dosen pembimbing) untuk menentukan apakah mereka benar-benar mirip atau tidak. Hasil penilaian manual ini akan menjadi *ground truth*.
2. Skenario Pengujian:
   - Setiap judul dalam data uji akan dibandingkan dengan judul-judul lain (atau judul referensi) menggunakan metode cosine similarity yang telah diimplementasikan.
   - Berdasarkan skor cosine similarity dan threshold yang telah ditentukan, sistem akan memprediksi apakah pasangan judul tersebut mirip atau tidak.
3. Pengumpulan Hasil Prediksi:
   - Hasil prediksi dari metode (mirip/tidak mirip) akan dicatat untuk setiap pasangan dalam data uji.

E. Teknik Analisis

1. Perhitungan Similarity

   Langkah pertama dalam proses evaluasi adalah menghitung kemiripan (similarity) antar judul skripsi atau teks menggunakan metode *cosine similarity*. Cosine similarity mengukur seberapa mirip dua vektor dalam ruang multidimensi, dalam hal ini representasi teks atau fitur. Nilai similarity berkisar antara 0 hingga 1, di mana nilai mendekati 1 menunjukkan kemiripan yang sangat tinggi. Dalam konteks ini, setiap pasangan judul skripsi dibandingkan untuk melihat kemirip judul skripsi berdasarkan nilai cosine similarity.

2. Analisis Hasil Metrik

   Setelah nilai similarity dihitung, langkah berikutnya adalah membuat label ground truth. Ground truth adalah data pembanding (label sebenarnya) yang digunakan untuk mengevaluasi akurasi hasil prediksi. Dalam proses ini, digunakan ambang batas (threshold) tertentu, misalnya 0.5. Jika nilai similarity antara judul skrispi sama dengan atau lebih besar dari threshold ini, maka pasangan tersebut diberi label "similar". Sebaliknya, jika nilainya di bawah threshold, pasangan tersebut dianggap "tidak similar". Ground truth ini menjadi referensi untuk mengukur kinerja sistem prediksi.

3. Diskusi dan Interpretasi

   Untuk mengetahui sensitivitas sistem terhadap tingkat kemiripan yang berbeda, dilakukan pengujian dengan menggunakan beberapa nilai threshold, misalnya 0.6, 0.7, dan 0.8. Setiap nilai threshold ini menghasilkan prediksi yang berbeda—semakin tinggi threshold, semakin ketat sistem dalam menganggap dua judul sebagai "mirip". Dengan mencoba berbagai ambang batas, peneliti dapat mengevaluasi performa sistem pada skenario yang berbeda dan memilih threshold yang paling sesuai dengan kebutuhan aplikasi.

4. Perhitungan Metrics

Tahap terakhir adalah mengevaluasi performa sistem prediksi dengan membandingkan hasil prediksi terhadap ground truth. Pengukuran dilakukan dengan menghitung metrik evaluasi umum seperti accuracy (tingkat ketepatan klasifikasi), precision (ketepatan prediksi positif), recall (kemampuan mendeteksi seluruh data positif), dan f1-score (rata-rata harmonis antara precision dan recall). Dengan metrik-metrik ini, performa sistem dapat dianalisis secara komprehensif untuk menentukan seberapa baik sistem membedakan judul yang mirip dan tidak mirip.

BAB IV
HASIL DAN PEMBAHASAN

A. Hasil Implementasi

1. Pengumpulan Data

Langkah pertama dalam penelitian ini adalah memperoleh data judul skripsi dari tabel judul_skripsi pada database MySQL. Data yang berhasil dikumpulkan terdiri atas dua atribut utama, yakni nomor identitas dan judul skripsi, dengan total sebanyak 1000 judul. Selama proses ekstraksi, dilakukan pengecekan agar setiap baris data benar-benar sesuai dan dapat digunakan untuk proses analisis selanjutnya. Data ini nantinya akan menjadi sumber utama yang diolah dalam tahap pra-pemrosesan serta pelatihan model untuk mengukur tingkat kemiripan antar judul skripsi.

1	penulis	judul_skripsi
2	A AHMAD ZULFIKAR	Pengaruh Tingkat Pendidikan Orang Tua Terhadap Pembentukan Kepribadian Anak (Studi Komunikasi Dalam Keluarga Pada Lingkungan Caile Kabupaten Sinjai)
3	A FAJRUL HAQ	BUDAYA KONSUMTIF MASYARAKAT SINJAI (STUDI KASUS PKL LAPANGAN SINJAI BERSATU)
4	A FITRIANI	Pengaruh media gambar terhadap minat baca anak usia dini di TK Insan Cemerlang Mannuruki Makassar
5	A INDAHSARY	TINDAK TUTUR PAPPAISSENG MASYARAKAT KAJUARA KABUPATEN BONE
6	A INDAHSARY	TINDAK TUTUR PAPPAISSENG MASYARAKAT KAJUARA KABUPATEN BONE
7	A ISMAYANTI	PERANAN PENDIDIK PKN DALAM MENUMBUHKAN SIKAP NASIONALISME SISWA SMA NASIONAL MAKASSAR
8	A MUH HADRIYANTO	EFEKTIVITAS PENYULUHAN PERTANIAN TERHADAP PERUBAHAN SOSIAL EKONOMI MASYARAKAT KAJUARA KABUPATEN BONE
9	A NUR INAYAH RAKHMAD	PENGARUH PENGGUNAAN METODE DISCOVERY BERBASIS MEDIA REALITA TERHADAP HASIL BELAJAR IPA KONSEP SIFAT DAN PERUBAHAN WUJUD BENDA SISWA KELAS IV SDN 8 KALIMBAUNG KECAMATAN BANTAENG KABUPATEN BANTAENG
10	A NURDAWANI	Pengaruh Penguasaan Bahasa indonesia terhadap Kemampuan Menyelesaikan Soal Cerita Matematika Siswa Kelas V SD Inpres Bontomanai Kota Makassar
11	A RESKI AMALIA YUSMAN	PENERAPAN METODE SAINTIFIK BERBASIS LITERASI SAINS TERHADAP KETERAMPILAN PROSES SAINS PESERTA DIDIK MA MUALLIMIN MUHAMMADIYAH MAKASSAR
12	A RESKIANTI ARDI	Efektivitas Pembelajaran Matematika melalui Penerapan Pendekatan Realistic Mathematics Education (RME) pada Siswa Kelas VIII A SMP Unismuh Makassar
13	A RUSGANDI SURYA NINGRAT	PENGEMBANGAN USAHA DAN PROBLEMA MASYARAKAT DI DALAM KAWASAN HUTAN KECAMATAN SIMBORO KABUPATEN MAMUJU STUDY KASUS MASYARAKAT DUSUN TABUNIA
14	A SERLY ANGRENI	PENGARUH METODE KOOPERATIF TIPE THINK PAIR SHARE (TPS) TERHADAP HASIL BELAJAR FISIKA PADA PESERTA DIDIK MAN BANTAENG
15	A SRI ATIRA	THE USE OF TALKING CHIPS METHOD IN IMPROVING STUDENTS' SPEAKING SKILL OF SMPN 39 BULUKUMBA
16	A SYAMSUL BAHRI	Perilaku sosial remaja dalam menggunakan ruang publik perkotaan (studi kasus Benteng Rotterdam kota makassar)
17	A UMMUL HAIFA	PENGARUH PENERAPAN METODE STORYTELLING TERHADAP HASIL BELAJAR KETERAMPILAN BERBICARA PADA MATA PELAJARAN BAHASA INDONESIA SISWA KELAS V SD INPRES PULLAWENG KECAMATAN EREMERASA KABUPATEN BANTAENG
18	A WAHYUNI	KULTUR PESANTREN DALAM PEMBINAAN KARAKTER SANTRI DI PONDOK PESANTREN DARUL ARQAM MUHAMMADIYAH GOMBARA MAKASSAR.
19	A YUNIARTI DIAN PRATIWI	THE EFFECT OF MNEMONIC TECHNIQUE WITH CROSSWORD PUZZLE MEDIA IN TEACHING VOCABULARY TO LEARNERS AT SMAN 11 WATAMPONE (A PRE-EXPERIMENTAL RESEARCH)
20	A. ARIKA WIBOWO	EKSISTENSI KOMUNITAS LITERASI UNIVERSITAS MUHAMMADIYAH MAKASSAR (SUATU KAJIAN RELASI KUASA MICHEL FOUCAULT)
21	A. ASRIANI	Efektivitas Pembelajaran Matematika Melalui Pendekatan Kontekstual Pada Siswa Kelas VII.2 SMP Negeri 4 Bajeng Kabupaten Gowa
22	A. HIKMA WARDANI	Penerapan Metode Eksperimen Pada Pembelajaran Fisika Terhadap Penigkatan Hasil Belajar Peserta Didik Kelas XI SMA Negeri 13 Luwu
23	A. HIKMA WARDANI	Penerapan Metode Eksperimen Pada Pembelajaran Fisika Terhadap Penigkatan Hasil Belajar Peserta Didik Kelas XI SMA Negeri 13 Luwu
24	A. HILYATUL MUSLIMAT	EFEKTIVITAS PEMBELAJARAN MATEMATIKA MELALUI PENERAPAN METODE KUMON PADA SISWA KELAS XI TKJ SMK MUHAMMADIYAH 4 TALLO
25	A. MARJULI TENRIOLA, A. M	EFEKTIVITAS PEMBELAJARAN MATEMATIKA DENGAN MENGGUNAKAN MODEL KOOPERATIF TIPE NUMBERED HEADS TOGETHER (NHT) PADA SISWA KELAS VII SMP NEGERI 2 PANGKAJENE
26	A. MUNAWARA	EFEKTIVITAS PEMBELAJARAN MATEMATIKA MELALUI PENERAPAN MODEL KOOPERATIF TIPE NUMBERED HEADS TOGETHER (NHT) PADA SISWA KELAS XI IPA SMA MUHAMMADIYAH WILAYAH MAKASSAR
27	A. MUSDALIFAH	Deskripsi pembelajaran matematika melalui penerapan model Quantum Teaching pada siswa kelas VIIc SMP N 1 Pallangga kabupaten Gowa
28	A. NURASIFAH HASYIM	PENGARUH PENERAPAN METODE STRUKTURAL ANALITIK SINTETIK (SAS) TERHADAP KEMAMPUAN MEMBACA PERMULAAN SISWA KELAS I SD NEGERI 167 KASUSO KECAMATAN BONTOBAHARI KABUPATEN BULUKUMBA
29	A. NURASIFAH HASYIM	PENGARUH PENERAPAN METODE STRUKTURAL ANALITIK SINTETIK (SAS) TERHADAP KEMAMPUAN MEMBACA PERMULAAN SISWA KELAS I SD NEGERI 167 KASUSO KECAMATAN BONTOBAHARI KABUPATEN BULUKUMBA
30	A. NURFAJRIANA	EFEKTIVITAS PEMBELAJARAN MATEMATIKA MELALUI PENERAPAN MODEL RECIPROCAL TEACHING SETTING KOOPERATIF PADA SISWA KELAS VII SMP NEGERI 15 BULUKUMBA
31	A. NURINDRA Y	ANALISIS PROGRAM PENGUATAN PENDIDIKAN KARAKTER PADA MATA PELAJARAN PENDIDIKAN AGAMA ISLAM PAI DI SD NEGERI UNGGULAN BONTOMANAI KECAMATAN BONTOMARANNU KABUPATEN GOWA
32	A. ROSDIANA AMIR	MAKNA SOSIAL SUMANGE' TEALARA TERHADAP IMPLEMENTASI MOTIVASI HIDUP SINGLE PARENT (STUDI KASUS DESA LABUAJA KECAMATAN KAHU KABUPATEN BONE)
33	A. SURYANINGSIH ARIYA	Perbandingan Hasil Belajar Matematika Antara Siswa yang Pembelajarannya melalui Model Kooperatif Tipe Times Games Tournament (TGT) dengan Tipe Numbered Heads Together (NHT) Pada Kelas VIIII SMP Somba Opu
34	A.RATNA PRATIWI PUTRI	HUBUNGAN POLA ASUH ORANG TUA DENGAN KEMANDIRIAN ANAK USIA 5-6 TAHUN DI TK ANNISA KABUPATEN KEPULAUAN SELAYAR PROVINSI SULAWESI SELATAN
35	A.WAHYUNI IRMAZANA	ANALISIS KESALAHAN MENYELESAIKAN SOAL CERITA MATERI GEOMETRI BERDASARKAN TEORI NEWMAN PADA KELAS VIII SMP NEGERI 3 SUNGGUMINASA
36	AAN INDRAYADI	STUDENTS' ABILITY IN IDENTIFYING INDEPENDENT CLAUSE AND DEPENDENT CLAUSE IN COMPLEX SENTENCE (DESCRIPTIVE STUDY AT THE SECOND SEMESTER STUDENTS OF ENGLISH DEPARTMENT UNISMUH MAKASSAR)
37	ABD QADIIR JAELANI	Efektivitas Pembelajaran Matematika Melalui Penerapan Model Kooperatif Tipe Talking Stick pada Siswa Kelas VII 1 SMP Negeri 1 Galesong Utara
38	ABD RAHMAN	Hubungan Pengelolaan Kelas Terhadap Motivasi Belajar Murid SDN Gentungang Kecamatan Bajeng Barat Kabupaten Gowa
39	ABD RAHMAN	PENINGKATAN KEMAMPUAN MENULIS KARANGAN NARASI MENGGUNAKAN STRATEGI AKTIVITAS TERBIMBING MURID KELAS V SEKOLAH DASAR INPRES 170 KAPASA DESA KASSI KECAMATAN RUMBIA KABUPATEN JENEPONTO
40	ABD RAHMAN	PENINGKATAN KEMAMPUAN MENULIS KARANGAN NARASI MENGGUNAKAN STRATEGI AKTIVITAS TERBIMBING MURID KELAS V SEKOLAH DASAR INPRES 170 KAPASADA DESA KASSI KECAMATAN RUMBIA KABUPATEN JENEPONTO
41	ABD RAHMAN	[illegible]

Gambar 4. Judul skripsi hasil ekstraksi dari database.

2. Preprocessing Data

Pra-pemrosesan data merupakan tahapan awal untuk memastikan data yang digunakan sudah bersih dan seragam. Salah satu proses utama dalam tahap ini adalah mengkonversi seluruh teks judul skripsi menjadi huruf kecil.

Cara ini diterapkan untuk menghindari masalah duplikasi akibat perbedaan penulisan huruf kapital, sehingga proses analisis dapat berjalan lebih konsisten dan hasil yang diperoleh menjadi lebih valid. Tahapan ini termasuk dalam proses Natural Language Processing (NLP) yang bertujuan meningkatkan kualitas data sebelum dianalisis lebih lanjut.

Melalui tahapan ini, seluruh judul skripsi yang digunakan dalam penelitian telah memiliki format yang seragam dan siap untuk dianalisis pada tahap selanjutnya, yaitu penghitungan tingkat kemiripan.

20	PENERAPAN MODEL INQUIRI TERBIMBING UNTUK MENINGKATKAN HASIL BELAJAR IPA KELAS V SDN LEMBAYA KECAMATAN TOMPOBULU KABUPATEN GOWA
21	PENINGKATAN HASIL BELAJAR MATEMATIKA DENGAN PENDEKATAN REALISTIK MATEMATIC EDUCATION (RME) PADA MURID KELAS IV SDN NO 1 CENTRE PATALLASSANG KECAMAT
22	PENINGKATAN HASIL BELAJAR IPS MELALUI MODEL PEMBELAJARAN OUTDOOR LEARNING MURID KELAS V SD NEGERI NO. 18 MAERO KECAMATAN BONTORAMBA KABUPATEN JEN
23	PENINGKATAN KEMAMPUAN MENULIS KARANGAN DESKRIPSI MELALUI TEKNIK MIND MAPPING MURID KELAS II SDN NO 197 INP BONTOPAJJA KECAMATAN GALESONG UTARA KA
24	KESANTUNAN BERBAHASA INDONESIA SISWA TERHADAP GURU PADA PROSES KEGIATAN PEMBELAJARAN DARING DI MASA PANDEMI SISWA KELAS VII SMPN SATAP PUNAGA KAB
25	IMPLEMENTASI KEBIJAKAN SEKOLAH RUJUKAN SMP NEGERI 1 SUNGGUMINASA KABUPATEN GOWA
26	EFEKTIFITAS PENERAPAN MODEL PEMBELAJARAN PROBLEM BASED LEARNING TERHADAP KEMAMPUAN BERPIKIR KRITIS SISWA PADA MATA PELAJARAN PPKN KELAS V SD NEGERI
27	PENGEMBANGAN MEDIA PEMBELAJARAN FISIKA BERBASIS ADOBE AIR FOR ANDROID SEBAGAI DAYA DUKUNG PEMBELAJARAN PESERTA DIDIK
28	PENERAPAN MODEL PEMBELAJARAN THINK PAIR SHARE (TPS) DALAM MENINGKATKAN HASIL BELAJAR IPS SISWA KELAS V SD NEGERI PANCIRO KECAMATAN BAJENG KABUPATEN
29	PENGARUH PENGGUNAAN MODEL PROBLEM CENTERED LEARNING (PCL) TERHADAPA KEMAMPUAN MENULIS NARASI SISWA KELAS V SDN 136 SALOBUNDANG KECAMATAN BON
30	PENGARUH KEPEMIMPINAN GURU TERHADAP MOTIVASI BELAJAR PESERTA DIDIK DI SD NEGERI 9 BANUA KABUPATEN MAJENE
31	ANALISIS TINGKAT KESULITAN SISWA DALAM MENULIS CERITA PADA MATA PELAJARAN BAHASA INDONESIA DI KELAS IV SDN MANNURUKI
32	ANALISIS KEMAMPUAN PEMAHAMAN KONSEP MATEMATIKA PADA MATERI GARIS DAN SUDUT DITINJAU DARI GAYA KOGNITIF SISWA KELAS VII SMPN 5 PALLANGGA
33	WHATSAPP SEBAGAI MEDIA PEMBELAJARAN PADA MASA PANDEMI COVID-19 DI SDN 346 TIMBULA KECAMATAN BONTO TIRO KABUPATEN BULUKUMBA
34	ANALISIS KETERAMPILAN BERPIKIR KRITIS SISWA SELAMA PEMBELAJARAN FISIKA SECARA DARING DI SMA MUHAMMADIYAH 9 MAKASSAR
35	PENGARUH PENGGUNAN MEDIA KOTAK HURUF TERHADAP KEMAMPUAN MEMBACA PERMULAAN SISWA KELAS 1 SDN KARANGLOE KECAMATAN TOMPOBULU KABUPATEN GOWA
36	TIPOLOGI SOLIDARITAS SOSIAL PETANI PENGGARAP DENGAN PETANI BESAR DI KABUPATEN MAROS (TINJAUAN TEORI PERTUKARAN SOSIAL)
37	ANALISIS PENGUNAAN MEDIA PEMBELAJARAN BERBASIS POWER POINT SPARKOL VIDEOSCRIBE TERHADAP MINAT BELAJAR SISWA PADA MATA PELAJARA BIOLOGI KELAS VII SMP I
38	PROSES BERPIKIR SISWA DALAM PEMECAHAN MASALAH MATEMATIKA BERDASARKAN GAYA KOGNITIF PADA SISWA KELAS VIII SMP NEGERI 1 BAJENG
39	PENGARUH PENGGUNAAN MEDIA BIG BOOK TERHADAP HASIL BELAJAR IPS SISWA KELAS IVB SD INPRES PAKU KECAMATAN PALLANGGA KABUPATEN GOWA
40	PENERAPAN SIKLUS BELAJAR KAROLUS TERHADAP KETERAMPILAN PROSES SAINS PESERTA DIDIK MADRASAH ALIYAH BONTOMARANNU
41	DESKRIPSI KEMAMPUAN SISWA DALAM MEMECAHKAN MASALAH GEOMETRI BERDASARKAN TEORI VAN HIELE PADA KELAS VIIIA SMPN 21 MAKASSAR
42	PENGEMBANGAN INSTRUMEN BERBASIS KETERAMPILAN BERPIKIR KREATIF PESERTA DIDIK PADA MATERI FLUIDA STATIS DI SMA NEGERI 15 GOWA
43	ANALISIS KEMAMPUAN REPRESENTASI MATEMATIS PADA MATERI HIMPUNAN DI TINJAU DARI GAYA KOGNITIF SISWA KELAS VII SMP NEGERI 5 MAKASSAR
44	DESKRIPSI KEMAMPUAN BERPIKIR KRITIS MATEMATIS SISWA DALAM MENYELESAIKAN SOAL MATEMATIKA MATERI RELASI DAN FUNGSI DITINJAU DARI PERBEDAAN GENDER PADA
45	EFEKTIVITAS PEMBELAJARAN MATEMATIKA MELALUI PENDEKATAN ELPSA PADA PESERTA DIDIK KELAS VIII A SMP NEGERI 1 GALESONG UTARA

Gambar 5. Data sebelum preprocessing

20	penerapan model inquiri terbimb meningkatkan hasil belajar ipa kela v sdn lembaya kecamatan tompobulu kabupaten gowa
21	peningkatan hasil belajar matematika pendekatan realistik matemat educ rme murid kela iv sdn no centr pitailassang kecamatan patallassang kabupaten takalar
22	peningkatan hasil belajar ip model pembelajaran outdoor learn murid kela v sd negeri no maero kecamatan bontoramba kabupaten jeneponto
23	peningkatan kemampuan menuli karangan deskripsi teknik mind map murid kela ii sdn no inp bontopajja kecamatan galesong utara kabupaten takalar
24	kesantunan berbahasa indonesia siswa guru prose kegiatan pembelajaran dare pandemi siswa kela vii smpn satap punaga kabupaten takalar
25	implementasi kebijakan sekolah rujukan smp negeri sungguminasa kabupaten gowa
26	efektifita penerapan model pembelajaran problem base learn kemampuan berpikir kriti siswa mata pelajaran ppkn kela v sd negeri dena
27	pengembangan media pembelajaran fisika berbasi adob air for android daya dukung pembelajaran peserta didik
28	penerapan model pembelajaran think pair share tp meningkatkan hasil belajar ip siswa kela v sd negeri panciro kecamatan bajeng kabupaten gowa
29	pengaruh penggunaan model problem center learn pcl terhadapa kemampuan menuli narasi siswa kela v sdn salobundang kecamatan bontotiro kabupaten bulukumba
30	pengaruh kepemimpinan guru motivasi belajar peserta didik sd negeri banua kabupaten majen
31	analisi tingkat kesulitan siswa menuli cerita mata pelajaran bahasa indonesia kela iv sdn mannuruki
32	analisi kemampuan pemahaman konsep matematika materi gari sudut tinjau gaya kognitif siswa kela vii smpn pallangga
33	whatsapp media pembelajaran pandemi covid sdn timbula kecamatan bonto tiro kabupaten bulukumba
34	analisi keterampilan berpikir kriti siswa pembelajaran fisika dare sma muhammadiyah makassar
35	pengaruh penggunan media kotak huruf kemampuan membaca permulaan siswa kela sdn karanglo kecamatan tompobulu kabupaten gowa
36	tipolog solidarita sosial petani penggarap petani kabupaten maro tinjauan teori pertukaran sosial
37	analisi pengunaan media pembelajaran berbasi power point sparkol videoscrib minat belajar siswa mata pelajara biolog kela vii smp pgri bontoramba kec bontoramba
38	prose berpikir siswa pemecahan matematika berdasarkan gaya kognitif siswa kela viii smp negeri bajeng
39	pengaruh penggunaan media big book hasil belajar ip siswa kela ivb sd inpr paku kecamatan pallangga kabupaten gowa
40	penerapan siklu belajar karolu keterampilan prose sain peserta didik madrasah aliyah bontomarannu
41	deskripsi kemampuan siswa memecahkan geometri berdasarkan teori van hiel kela viiia smpn makassar
42	pengembangan instrumen berbasi keterampilan berpikir kreatif peserta didik materi fluida stati sma negeri gowa
43	analisi kemampuan representasi matemati materi himpunan tinjau gaya kognitif siswa kela vii smp negeri makassar
44	deskripsi kemampuan berpikir kriti matemati siswa menyelesaikan matematika materi relasi fungsi ditinjau perbedaan gender siswa kela viii smpn makassar
45	efektivita pembelajaran matematika pendekatan elpsa peserta didik kela viii a smp negeri galesong utara

Gambar 6. Data sesudah preprocessing

3. Ekstraksi Fitur (Representasi TF-IDF).

```
def extract_features(self):
    if not hasattr(self, 'processed_titles'):
        self.log("No processed data available")
        return

    self.tfidf_matrix = self.tfidf_vectorizer.fit_transform(self.
    processed_titles)
    self.log(f"Feature extraction completed. TF-IDF matrix shape: {self.
    tfidf_matrix.shape}")
```

Setelah data judul skripsi selesai dibersihkan melalui proses *preprocessing*, tahap selanjutnya adalah melakukan ekstraksi fitur menggunakan metode TF-IDF (*Term Frequency - Inverse Document Frequency*). Tujuan dari tahap ini adalah mengubah kumpulan kata-kata pada judul skripsi menjadi bentuk angka atau nilai bobot, agar bisa dihitung oleh komputer. Secara sederhana, TF-IDF akan melihat seberapa sering sebuah kata muncul dalam satu judul (*term frequency*), namun juga mempertimbangkan apakah kata tersebut sering muncul di banyak judul lain (*inverse document frequency*). Dengan begitu, kata-kata yang sering muncul di satu judul tetapi jarang muncul di judul lain akan memiliki bobot yang lebih tinggi, karena dianggap lebih penting. Sebaliknya, kata-kata yang muncul hampir di semua judul (misalnya "analisis", "pengaruh") akan memiliki bobot lebih rendah. Hasil dari proses ini adalah sebuah vektor angka yang mewakili isi dari setiap judul skripsi. Vektor inilah yang nantinya digunakan untuk mengukur kemiripan antar judul pada tahap selanjutnya.

4. Implementasi Perhitungan Cosine Similarity

```
def detect_similarity(self, threshold=0.6):
    if self.tfidf_matrix is None:
        self.log("No TF-IDF matrix available")
        return []

    cosine_sim = cosine_similarity(self.tfidf_matrix, self.tfidf_matrix)


    similar_pairs = []


    for i in range(len(cosine_sim)):
        for j in range(i+1, len(cosine_sim)):
            similarity = cosine_sim[i][j]

            author1 = str(self.ids[i]).split('_')[0] if '_' in str(self.ids
            [i]) else str(self.ids[i])
            author2 = str(self.ids[j]).split('_')[0] if '_' in str(self.ids
            [j]) else str(self.ids[j])


            if similarity ≥ 0.999 and author1 == author2:
                continue

            if similarity ≥ threshold:
                similar_pairs.append({
                    'id1': self.ids[i],
                    'title1': self.titles[i],
                    'id2': self.ids[j],
                    'title2': self.titles[j],
                    'similarity': float(similarity)
                })
    return similar_pairs
```

Setelah setiap judul skripsi diubah menjadi vektor angka menggunakan TF-IDF, tahap berikutnya adalah menghitung tingkat kemiripan antar judul menggunakan rumus cosine similarity. Cara kerjanya adalah dengan membandingkan arah vektor dari judul baru dengan vektor-vektor judul yang sudah ada sebelumnya. Hasil perhitungan cosine similarity ini berupa angka antara 0 sampai 1, di mana nilai 1 berarti sangat mirip (sudut vektornya sama), sedangkan nilai 0 berarti tidak mirip sama sekali (sudut vektornya 90 derajat). Semakin mendekati angka 1, maka kedua judul tersebut dianggap semakin mirip isi atau temanya. Proses ini sangat membantu untuk mendeteksi apakah ada judul baru yang terlalu mirip dengan judul skripsi yang sudah pernah ada.

5. Penentuan Threshold (Ambang Batas)

   Setelah program menghitung skor kemiripan antar judul menggunakan cosine similarity, langkah berikutnya adalah menentukan threshold atau ambang batas. Ambang batas ini digunakan untuk memutuskan apakah sebuah judul skripsi baru dianggap mirip dengan judul yang sudah ada atau tidak. Biasanya, nilai threshold ini dipilih berdasarkan pengalaman atau percobaan, misalnya 0.5 atau 0.7. Artinya, jika skor kemiripan antara judul baru dan judul lama lebih besar atau sama dengan threshold, maka program akan menganggap judul tersebut terlalu mirip atau bahkan duplikat. Sebaliknya, jika nilainya di bawah threshold, maka judul dianggap berbeda dan aman untuk dipakai. Threshold ini penting agar program tidak terlalu sensitif (semua dibilang mirip) atau terlalu longgar (judul duplikat malah dianggap berbeda).

6. Proses Utama Aplikasi

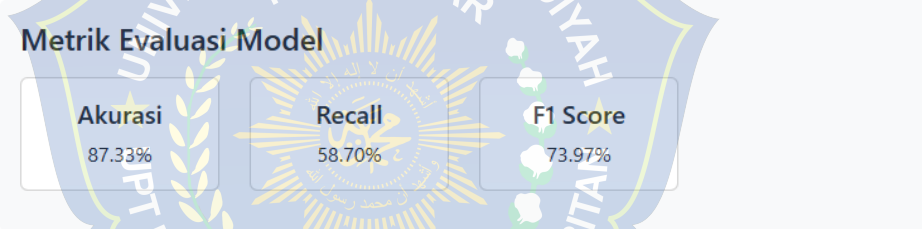
   a. Input Judul

   Tampilan yang ditunjukkan pada gambar adalah antarmuka dari sebuah sistem yang berfungsi untuk mendeteksi kemiripan judul skripsi. Sistem ini dirancang untuk membantu pengguna, khususnya mahasiswa atau dosen, dalam memeriksa apakah judul skripsi yang diusulkan memiliki kesamaan dengan judul-judul yang telah ada sebelumnya. Tujuan dari

fitur ini adalah untuk menghindari duplikasi judul dan memastikan orisinalitas karya ilmiah yang akan dibuat.

Pada bagian atas tampilan terdapat judul utama yang berbunyi "Deteksi Kemiripan Judul Skripsi". Judul ini memberikan informasi yang jelas mengenai fungsi dari sistem yang sedang digunakan. Tepat di bawahnya terdapat kalimat penjelas singkat yang mengarahkan pengguna untuk memasukkan judul skripsi yang ingin diperiksa tingkat kemiripannya. Sistem ini menyediakan sebuah kolom teks dengan label "Judul Skripsi" dan petunjuk input berupa placeholder yang bertuliskan "Masukkan judul skripsi yang ingin diperiksa...". Kolom ini digunakan oleh pengguna untuk mengetikkan judul skripsi yang akan dianalisis. Setelah itu, pengguna dapat menekan tombol biru bertuliskan "Periksa Kemiripan" untuk memulai proses pemeriksaan.

b. Metrik Evaluasi Model



Gambar tersebut menampilkan hasil evaluasi dari kinerja model yang digunakan untuk mendeteksi kemiripan judul skripsi. Evaluasi ini disajikan melalui tiga metrik utama, yaitu akurasi, recall, dan F1 score. Akurasi model sebesar 87,33% menunjukkan bahwa sebagian besar prediksi yang dilakukan oleh model sudah sesuai dengan data sebenarnya. Ini berarti model cukup handal dalam mengklasifikasikan apakah judul skripsi tergolong mirip atau tidak.

Namun, nilai recall yang hanya sebesar 58,70% mengindikasikan bahwa model masih melewatkan beberapa judul yang seharusnya terdeteksi sebagai mirip. Recall sendiri menggambarkan kemampuan model dalam

menemukan semua data yang relevan, sehingga semakin tinggi nilai recall, semakin baik sensitivitas model terhadap judul-judul yang mirip. Sementara itu, nilai F1 score sebesar 73,97% menunjukkan keseimbangan antara presisi dan recall. Nilai ini cukup baik, yang berarti model tidak hanya cukup akurat dalam hasilnya, tetapi juga relatif seimbang dalam mendeteksi dan menghindari kesalahan

7. Hasil Pengujian

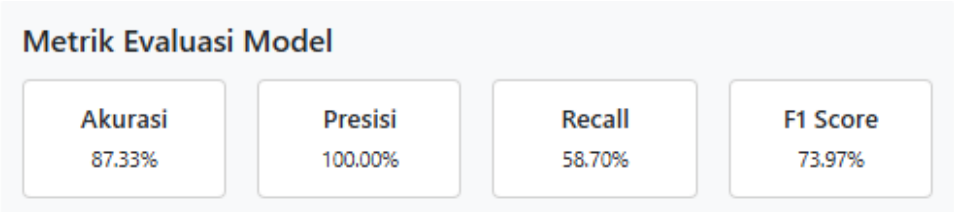
Pengujian sistem dilakukan dengan cara menginput salah satu judul skripsi, yaitu "Pengembangan e-komik media pembelajaran menulis teks negosiasi". Judul ini dipilih untuk menguji kemampuan sistem dalam mendeteksi kemiripan dengan judul-judul skripsi lain yang telah tersimpan dalam basis data. Setelah proses input dilakukan, sistem secara otomatis mengonversi judul tersebut ke dalam representasi vektor melalui metode TF-IDF, kemudian menghitung skor kemiripan dengan seluruh judul yang ada di dalam database menggunakan pendekatan Cosine Similarity.

Sebagai hasil dari proses tersebut, sistem menghasilkan daftar judul skripsi yang dianggap memiliki tingkat kemiripan tertentu dengan judul yang diuji. Lima judul dengan skor kemiripan tertinggi ditampilkan kepada pengguna, lengkap dengan persentase kemiripannya. Sebagai contoh, judul dengan skor tertinggi adalah “Peningkatan Pembelajaran Menulis Teks Prosedur Kompleks dengan Menggunakan Pendekatan Proses bagi Siswa SMK Negeri 2 Bungoro” yang memperoleh skor kemiripan sebesar 34.7%. Judul lainnya seperti “Keefektifan Model Pembelajaran Berbasis Proyek pada Pembelajaran Menulis Teks Negosiasi” dan “Efektivitas Implementasi Model Pembelajaran Berbasis Masalah pada Teks Eksposisi” juga terdeteksi dengan skor mendekati 30%. Hal ini membuktikan bahwa sistem mampu menangkap hubungan semantik antarjudul meskipun tidak identik secara leksikal.

Gambar 4.1: Hasil Deteksi Kemiripan Judul.

Selanjutnya, untuk mengevaluasi performa model secara menyeluruh, digunakan empat metrik utama, yaitu akurasi, presisi, recall, dan F1 score. Berdasarkan hasil evaluasi, diperoleh akurasi sebesar 87.33%, yang menunjukkan bahwa sebagian besar prediksi sistem telah sesuai dengan data yang sebenarnya. Nilai presisi mencapai 100.00%, menandakan bahwa semua prediksi kemiripan yang dibuat oleh sistem benar-benar tepat tanpa adanya kesalahan klasifikasi positif palsu (false positive). Namun demikian, recall sistem masih berada pada angka 58.70%, yang berarti masih terdapat sejumlah judul mirip yang belum berhasil terdeteksi.

Untuk mengukur keseimbangan antara presisi dan recall, digunakan metrik F1 Score yang berada pada nilai 73.97%, menunjukkan performa model berada pada tingkat yang cukup baik.



Gambar 4.2: Metrik Evaluasi Model

Secara keseluruhan, hasil pengujian ini menunjukkan bahwa sistem deteksi kemiripan judul skripsi berbasis pendekatan semantik TF-IDF dan Cosine Similarity telah mampu berfungsi dengan baik, khususnya dalam hal presisi. Namun, perbaikan pada aspek recall masih diperlukan agar sistem dapat mengenali lebih banyak judul yang memiliki kemiripan semantik secara menyeluruh. Dengan demikian, sistem ini sangat potensial untuk digunakan sebagai alat bantu dalam validasi keaslian judul skripsi dan pencegahan duplikasi topik di lingkungan akademik.

B. Pembahasan

Gambar yang ditampilkan sebelumnya menunjukkan hasil akhir dari implementasi sistem deteksi kemiripan judul skripsi yang telah dikembangkan dalam penelitian ini. Sistem ini dirancang untuk membantu pengguna, khususnya mahasiswa dan dosen pembimbing, dalam memverifikasi tingkat kemiripan judul skripsi baru terhadap database skripsi yang telah ada. Fitur ini berfungsi sebagai alat bantu dalam mencegah pengajuan judul yang bersifat duplikat atau terlalu mirip, sehingga dapat mendorong terciptanya orisinalitas dalam penulisan karya ilmiah di lingkungan akademik.

Melalui antarmuka sistem, pengguna dapat memasukkan judul skripsi yang ingin diperiksa. Setelah dilakukan proses pemeriksaan, sistem akan menampilkan daftar judul-judul lain yang dianggap memiliki kemiripan, lengkap dengan persentase tingkat kemiripannya.

Dalam contoh hasil pengujian yang ditampilkan, beberapa judul memiliki tingkat kemiripan mulai dari 22% hingga 34%. Informasi ini dilengkapi dengan nama penulis dari masing-masing judul skripsi yang ditemukan. Nilai persentase tersebut dihitung menggunakan metode cosine similarity terhadap representasi vektor teks hasil dari algoritma TF-IDF. Semakin tinggi nilai persentase kemiripan, maka semakin besar kemungkinan kedua judul memiliki kesamaan dalam hal kata kunci, struktur kalimat, atau bahkan topik bahasan.

Di bagian bawah tampilan sistem, disajikan pula hasil evaluasi performa dari model klasifikasi yang digunakan. Evaluasi dilakukan dengan menggunakan empat metrik utama, yaitu akurasi, presisi, recall, dan F1 score. Hasil evaluasi menunjukkan bahwa akurasi model mencapai 87,33%, menandakan bahwa sebagian besar prediksi sistem sudah sesuai dengan kategori sebenarnya, baik saat mengidentifikasi judul yang mirip maupun tidak. Nilai presisi mencapai 100%, yang berarti semua judul yang diklasifikasikan sebagai mirip benar-benar relevan tanpa adanya kesalahan klasifikasi positif (false positive). Hal ini menunjukkan keandalan sistem dalam memberikan rekomendasi hasil yang benar-benar mirip.

Namun, recall sistem yang berada pada angka 58,70% mengindikasikan masih adanya beberapa judul yang sebenarnya mirip, tetapi tidak berhasil dikenali oleh sistem (false negative). Rendahnya nilai recall ini kemungkinan disebabkan oleh keterbatasan representasi semantik pada metode TF-IDF, yang cenderung berfokus pada frekuensi kata dan belum sepenuhnya mampu menangkap konteks kalimat secara utuh. Variasi dalam gaya bahasa, penggunaan sinonim, atau struktur kalimat yang kompleks juga menjadi tantangan dalam proses deteksi kemiripan berbasis kata. Oleh karena itu, meskipun sistem sudah sangat baik dalam menghindari kesalahan positif, masih terdapat ruang untuk meningkatkan sensitivitas sistem dalam mendeteksi seluruh kemiripan yang relevan.

Untuk mengevaluasi keseimbangan antara presisi dan recall, digunakan metrik F1 Score, yang dalam sistem ini tercatat sebesar 73,97%. Nilai ini mencerminkan bahwa sistem telah mencapai performa yang cukup baik secara keseluruhan. Akan tetapi, peningkatan pada aspek recall tetap menjadi fokus penting untuk pengembangan sistem di masa mendatang, agar sistem tidak hanya tepat dalam hasil yang diberikan, tetapi juga lengkap dalam cakupan deteksi kemiripan judul.

.

BAB V
PENUTUP

A. Kesimpulan

Berdasarkan hasil penelitian dan implementasi yang telah dilakukan, dapat disimpulkan hal-hal sebagai berikut:

1. Implementasi metode Cosine Similarity dalam mendeteksi kesamaan judul skripsi mahasiswa FKIP Unismuh Makassar berhasil dilakukan dengan menggunakan pendekatan representasi teks berbasis TF-IDF. Sistem yang dikembangkan mampu menerima input judul baru dari pengguna dan membandingkannya dengan koleksi judul skripsi yang telah ada dalam basis data. Proses perhitungan cosine similarity memberikan nilai kemiripan dalam bentuk persentase, sehingga pengguna dapat dengan mudah mengidentifikasi sejauh mana sebuah judul menyerupai judul-judul sebelumnya.
2. Kinerja metode Cosine Similarity telah dievaluasi menggunakan metrik akurasi, presisi, recall, dan F1-score. Hasil evaluasi menunjukkan bahwa sistem memiliki akurasi sebesar 87,33%, presisi 100%, recall 58,70%, dan F1-score 73,97%. Angka presisi yang sangat tinggi menunjukkan bahwa sistem mampu mengidentifikasi judul-judul yang benar-benar mirip tanpa memberikan hasil palsu (false positive). Namun, nilai recall yang masih di bawah 60% menunjukkan bahwa sistem belum sepenuhnya mampu menangkap semua judul yang sebenarnya mirip, yang mengindikasikan adanya ruang untuk peningkatan, khususnya dalam hal sensitivitas deteksi.

B. Saran

Berdasarkan kesimpulan di atas, penulis memberikan beberapa saran untuk pengembangan lebih lanjut:

1. Selama proses penelitian, penulis mengalami kendala dalam pengumpulan dan normalisasi data judul skripsi yang tidak terstandar dari berbagai sumber. Oleh karena itu, ke depan disarankan agar sistem pendukung dilengkapi dengan modul *preprocessing* otomatis untuk menyaring dan menyusun data judul yang tidak

konsisten. Hal ini penting agar proses pengujian sistem dapat dilakukan lebih akurat dan efisien.

2. Perlu dilakukan pengujian lebih lanjut dengan data dari berbagai program studi, guna menguji generalisasi dan fleksibilitas sistem. Dengan begitu, sistem tidak hanya terbatas pada FKIP Unismuh Makassar, tetapi bisa diterapkan pada lingkup yang lebih luas.

DAFTAR PUSTAKA

Anugrah, I. G. (2021). Penerapan Metode N-Gram dan Cosine Similarity Dalam Pencarian Pada Repositori Artikel Jurnal Publikasi. *Building of Informatics, Technology and Science (BITS)*, 3(3), 275–284.

Haruna Hanjas. (2024). *DETEKSI TINGKAT KEMIRIPAN JUDUL SKRIPSI PRODI INFORMATIKA MENGGUNAKAN METODE SUPPORT VECTOR MACHINE & NATURAL LANGUANGE PROCESSING*.

Prismadana, T. A. (2023). Aplikasi Ruang Tugas Dengan Deteksi Kemiripan Teks Pada Dokumen Tugas Menggunakan Cosine Similarity. *Jurnal Informatika Dan Multimedia*, *15*(1). https://doi.org/10.33795/jtim.v15i1.4405

Rahmadianti, Z. A., Priharsari, D., & Perdanakusuma, A. R. (2023). Analisis Persepsi Dosen Terhadap Kebijakan Penggunaan Turnitin untuk Mendeteksi Plagiarisme Skripsi Mahasiswa. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *7*(3).

Soyusiawaty, D., & Jones, A. H. S. (2020). Pemanfaatan Bahasa Alami Dalam Penelusuran Informasi Skripsi Melalui Digital Library. *Mobile and Forensics*, *2(1)*. https://doi.org/10.12928/mf.v2i1.2040

Hasanaha, U. N., Satra, R., & Umar, F. (2020). Deteksi Kemiripan Judul Skripsi Menggunakan Algoritma Smith Waterman. Buletin Sistem Informasi dan Teknologi Islam, 1(1), 56-65.

Apriani, H., Zakiyudin, H., & Marzuki, K. (2023). Penerapan Algoritma Cosine Similarity dan Pembobotan TF-IDF System Penerimaan Mahasiswa Baru pada Kampus Swasta. Universitas Bumigora.

Prasetyo, V. R., Benarkah, N., & Chrisintha, V. J. (2021). Implementasi Natural Language Processing Dalam Pembuatan Chatbot Pada Program Information

Technology Universitas Surabaya. Program Studi Teknik Informatika, Universitas Surabaya, Surabaya, Jawa Timur.

Arbiantono, M. W., & Ekohariadi. (2023). Pengembangan Aplikasi *Assessment Tool* Menggunakan Metode Cosine Semantic Similarity untuk *Automatic Scoring* Jawaban Tes Uraian pada Mata Pelajaran Basis Data di SMKN 1 Surabaya. Pendidikan Teknologi Informasi, Fakultas Teknik, Universitas Negeri Surabaya.

Ahmad, I., Borman, R. I., Caksana, G. G., & Fakhrurozi, J. (2023). Implementasi String Matching dengan Algoritma Boyer Moore untuk Menentukan Tingkat Kemiripan pada Pengajuan Judul Skripsi/TA Mahasiswa (Studi Kasus: Universitas XYZ). Sistem Informasi, Universitas Teknokrat Indonesia, Fakultas Teknik dan Ilmu Komputer, Universitas Teknokrat Indonesia.

Utami, N. W., & Putra, I. G. J. (2023). Text Mining Clustering untuk Pengelompokan Topik Dokumen Penelitian Menggunakan Algoritma K-Means dengan Cosine Similarity. Sistem Informasi Akuntansi, STMIK Primakara.

Sutikno, H., & Saniati. (2023). Implementasi Algoritma Cosine Similarity untuk Mendeteksi Kemiripan Topik Judul. Program Studi S1 Informatika, Fakultas Teknik Dan Ilmu Komputer, Universitas Teknokrat Indonesia.

Lindang, M. I., Jumaidil, J., & Aswin, R. (2022). Sistem Penentuan Kemiripan Antar Judul Skripsi Menggunakan Metode Cosine Similarity. *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, 11(01), 1–9. https://doi.org/10.35907/komputa.v11i01.728

Manullang, M., Erma, Z., Razali, M., Rini, R., Tampubolon, M., & Sitepu, E. (2021). Sosialisasi Penggunaan Aplikasi Turnitin Bagi Dosen Dalam Upaya Menghindari Plagiarisme. *Journal Liaison Academia and Society*, 1(3), 26–33.

Nurwanda, N., Suarna, N., & Prihartono, W. (2024). Penerapan NLP (Natural Language Processing) Dalam Analisis Sentimen Pengguna Telegram Di Playstore. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(2), 1841–1846.

Nasrullah, M. (2024). Sistem Deteksi Kemiripan Judul Tugas Akhir Menggunakan Metode TF-IDF dan Cosine Similarity di Universitas Ichsan Gorontalo. *Jurnal Teknologi Informasi dan Ilmu Komputer (JATIKOM)*, 11(1), 50–58. https://doi.org/10.35971/jatikom.v11i1.1903

Nugroho, D. R., Ramadhan, R. R., & Khusaini, M. A. (2021). Sistem Informasi Manajemen Proposal Penelitian dan PKM Menggunakan Metode Cosine Similarity. *Jurnal Tekno Kompak*, 15(2), 103–111. https://doi.org/10.33365/tk.v15i2.1050

Pernanda, Y., & Hakiki, N. (2021). Deteksi Kemiripan Judul Skripsi Mahasiswa Menggunakan Algoritma Cosine Similarity. *Jurnal Media Informatika Budidarma*, 5(4), 1509–1514. https://doi.org/10.30865/mib.v5i4.2950

Zahwa, F. A., & Syafi'i, I. (2022). Pemilihan Pengembangan Media Pembelajaran Berbasis Teknologi Informasi. *Equilibrium: Jurnal Penelitian Pendidikan Dan Ekonomi*, 19(01), 61–78. https://doi.org/10.25134/equi.v19i01.3963

LAMPIRAN

Lampiran 1. *Source Code*

```
import os
import pandas as pd
import numpy as np
import mysql.connector
import re
import nltk
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from nltk.stem import PorterStemmer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity
from sklearn.model_selection import train_test_split, KFold
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
from dotenv import load_dotenv
import random
import matplotlib
matplotlib.use('Agg')
import matplotlib.pyplot as plt
import seaborn as sns
import io
import base64
from datetime import datetime
from flask import Flask, render_template, request, redirect, url_for, flash, jsonify

# Download NLTK resources
```

```
nltk.download('punkt_tab')
nltk.download('stopwords')

# Load environment variables
load_dotenv()

app = Flask(__name__)
app.secret_key = os.urandom(24)

class ThesisSimilarityDetector:
    def __init__(self):
        self.db_config = {
            'host': os.getenv('DB_HOST'),
            'user': os.getenv('DB_USER'),
            'password': os.getenv('DB_PASSWORD'),
            'database': os.getenv('DB_NAME'),
            'port': os.getenv('DB_PORT')
        }
        self.stemmer = PorterStemmer()
        self.stop_words = set(stopwords.words('indonesian'))
        self.tfidf_vectorizer = TfidfVectorizer()
        self.titles = None
        self.tfidf_matrix = None
        self.evaluation_results = None
        self.similar_pairs = None
        self.processed = False
        self.logs = []

    def collect_data(self):
```

```
"""Retrieve thesis titles from database"""
try:
    conn = mysql.connector.connect(**self.db_config)
    cursor = conn.cursor()

    # Fetch data from the fkip table (limit to 1000 records)
    query = "SELECT * FROM fkip LIMIT 1000"
    cursor.execute(query)

    # Get column names
    columns = [desc[0] for desc in cursor.description]

    # Fetch all rows
    rows = cursor.fetchall()

    # Create DataFrame
    df = pd.DataFrame(rows, columns=columns)

    # Check if there's a column for thesis titles
    title_column = None
    for possible_column in ['judul', 'judul_skripsi', 'title']:
        if possible_column in df.columns:
            title_column = possible_column
            break

    if title_column is None:
        raise ValueError("Could not find a column for thesis titles in the database")

    self.titles = df[title_column].tolist()
```

```
            self.ids = df.iloc[:, 0].tolist()  # Assuming first column is ID

            cursor.close()
            conn.close()

            self.log(f"Successfully collected {len(self.titles)} thesis titles")
            return df

        except Exception as e:
            self.log(f"Error collecting data: {e}")
            return None

    def preprocess_data(self):
        """Preprocess the thesis titles"""
        if not self.titles:
            self.log("No data to preprocess")
            return

        processed_titles = []

        for title in self.titles:
            # Convert to lowercase
            title = str(title).lower()

            # Remove special characters and numbers
            title = re.sub(r'[^\w\s]', '', title)
            title = re.sub(r'\d+', '', title)

            # Tokenize
```

```
        tokens = word_tokenize(title)

        # Remove stop words
        filtered_tokens = [word for word in tokens if word not in self.stop_words]

        # Stemming
        stemmed_tokens = [self.stemmer.stem(word) for word in filtered_tokens]

        # Join tokens back to string
        processed_title = ' '.join(stemmed_tokens)
        processed_titles.append(processed_title)

    self.processed_titles = processed_titles
    self.log("Data preprocessing completed")

def extract_features(self):
    """Extract features using TF-IDF"""
    if not hasattr(self, 'processed_titles'):
        self.log("No processed data available")
        return

    # Apply TF-IDF transformation
    self.tfidf_matrix = self.tfidf_vectorizer.fit_transform(self.processed_titles)
    self.log(f"Feature extraction completed. TF-IDF matrix shape:
    {self.tfidf_matrix.shape}")

def train_model(self):
    """Train the model (for SVM if used)"""
    # For Cosine Similarity, we don't need explicit model training
```

```
    # But we could implement SVM or other models here if needed
    self.log("Model preparation completed")

def detect_similarity(self, threshold=0.6):
    """Detect similarity between thesis titles"""
    if self.tfidf_matrix is None:
        self.log("No TF-IDF matrix available")
        return []

    # Compute cosine similarity matrix
    cosine_sim = cosine_similarity(self.tfidf_matrix, self.tfidf_matrix)

    # Create a list to store similar pairs
    similar_pairs = []

    # Check for similar pairs
    for i in range(len(cosine_sim)):
        for j in range(i+1, len(cosine_sim)):
            similarity = cosine_sim[i][j]
            # Extract author names from IDs if available
            author1 = str(self.ids[i]).split('_')[0] if '_' in str(self.ids[i]) else
str(self.ids[i])
            author2 = str(self.ids[j]).split('_')[0] if '_' in str(self.ids[j]) else
str(self.ids[j])

            # Skip if exact same title (similarity=1.0) AND same author
            if similarity >= 0.999 and author1 == author2:
                continue

            if similarity >= threshold:
```

```
                similar_pairs.append({
                    'id1': self.ids[i],
                    'title1': self.titles[i],
                    'id2': self.ids[j],
                    'title2': self.titles[j],
                    'similarity': float(similarity)
                })

    return similar_pairs

def create_evaluation_dataset(self):
    """Create a realistic evaluation dataset by generating ground truth labels"""
    if self.tfidf_matrix is None:
        self.log("No TF-IDF matrix available")
        return None

    # Compute cosine similarity matrix
    similarity_matrix = cosine_similarity(self.tfidf_matrix)
    np.fill_diagonal(similarity_matrix, 0)  # Set diagonal to 0

    # Create pairs and labels for evaluation
    all_pairs = []

    # Get all possible pairs
    for i in range(len(self.titles)):
        for j in range(i+1, len(self.titles)):
            # Store the pair and the actual similarity score
            all_pairs.append({
                'idx1': i,
```

```
                'idx2': j,
                'title1': self.titles[i],
                'title2': self.titles[j],
                'actual_similarity': similarity_matrix[i][j]
            })

    # Sample a balanced dataset for evaluation - total 1000 pairs
    # Choose high similarity pairs
    high_sim_pairs = [p for p in all_pairs if p['actual_similarity'] >= 0.7]
    if len(high_sim_pairs) > 333:
        high_sim_pairs = random.sample(high_sim_pairs, 333)

    # Choose medium similarity pairs
    med_sim_pairs = [p for p in all_pairs if 0.4 <= p['actual_similarity'] < 0.7]
    if len(med_sim_pairs) > 333:
        med_sim_pairs = random.sample(med_sim_pairs, 333)

    # Choose low similarity pairs
    low_sim_pairs = [p for p in all_pairs if p['actual_similarity'] < 0.4]
    if len(low_sim_pairs) > 334:
        low_sim_pairs = random.sample(low_sim_pairs, 334)

    # Combine all sampled pairs
    eval_pairs = high_sim_pairs + med_sim_pairs + low_sim_pairs
    random.shuffle(eval_pairs)

    # Assign ground truth labels
    for pair in eval_pairs:
        # Add some noise to make evaluation more realistic
```

```
            noise = random.uniform(-0.15, 0.15)
            adjusted_sim = pair['actual_similarity'] + noise

            # Use 0.5 as the threshold for ground truth labeling
            if adjusted_sim >= 0.5:
                pair['ground_truth'] = 1
            else:
                pair['ground_truth'] = 0

        self.log(f"Created evaluation dataset with {len(eval_pairs)} pairs")
        return eval_pairs

    def evaluate_model(self, thresholds=[0.5, 0.6, 0.7, 0.8]):
        """Evaluate the model using fixed train-test split"""
        if not hasattr(self, 'processed_titles') or self.tfidf_matrix is None:
            self.log("No processed data or TF-IDF matrix available")
            return None

        # Create evaluation dataset
        eval_pairs = self.create_evaluation_dataset()
        if not eval_pairs:
            return None

        # Extract features and labels
        X = [(p['idx1'], p['idx2']) for p in eval_pairs]
        y = [p['ground_truth'] for p in eval_pairs]

        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.9,
        random_state=42)
```

```
self.log(f"Split data into train ({len(X_train)} pairs) and test ({len(X_test)} pairs) sets")

# Store results for each threshold
threshold_results = {}

# Test multiple thresholds
for threshold in thresholds:
    self.log(f"\nEvaluating with similarity threshold: {threshold}")

    # Make predictions based on similarity scores and threshold
    y_pred = []
    for i, j in X_test:
        sim_score = cosine_similarity(
            self.tfidf_matrix[i].reshape(1, -1),
            self.tfidf_matrix[j].reshape(1, -1)
        )[0][0]
        y_pred.append(1 if sim_score >= threshold else 0)

    # Calculate metrics
    accuracy = accuracy_score(y_test, y_pred)
    precision = precision_score(y_test, y_pred, zero_division=0)
    recall = recall_score(y_test, y_pred, zero_division=0)
    f1 = f1_score(y_test, y_pred, zero_division=0)

    metrics = {
        'accuracy': accuracy,
        'precision': precision,
```

```
            'recall': recall,
            'f1_score': f1
        }

        self.log(f"Accuracy={accuracy:.4f}, Precision={precision:.4f}, "
            f"Recall={recall:.4f}, F1-Score={f1:.4f}")

        threshold_results[threshold] = metrics

    # Find the best threshold based on F1 score
    best_threshold = max(threshold_results, key=lambda t:
    threshold_results[t]['f1_score'])
    best_metrics = threshold_results[best_threshold]

    self.log(f"\nBest threshold: {best_threshold} with metrics:")
    for metric, value in best_metrics.items():
        self.log(f"{metric}: {value:.4f}")

    # Return the best threshold and its metrics
    return {
        'best_threshold': best_threshold,
        'metrics': best_metrics,
        'all_thresholds': threshold_results,
        'train_size': len(X_train),
        'test_size': len(X_test)
    }

def check_title_similarity(self, new_title):
    """Check similarity of a new title against existing titles"""
```

```
if not self.processed or self.tfidf_matrix is None:
    return {"error": "Model not trained yet. Please run the training process first."}

# Preprocess the new title
new_title = str(new_title).lower()
new_title = re.sub(r'[^\w\s]', '', new_title)
new_title = re.sub(r'\d+', '', new_title)
tokens = word_tokenize(new_title)
filtered_tokens = [word for word in tokens if word not in self.stop_words]
stemmed_tokens = [self.stemmer.stem(word) for word in filtered_tokens]
processed_title = ' '.join(stemmed_tokens)

# Transform using the existing vectorizer
new_title_vector = self.tfidf_vectorizer.transform([processed_title])

# Calculate similarity with all existing titles
similarities = cosine_similarity(new_title_vector, self.tfidf_matrix)[0]

# Get top 10 most similar titles
top_indices = similarities.argsort()[-10:][::-1]

results = []
for idx in top_indices:
    if similarities[idx] > 0:  # Only include if there's some similarity
        results.append({
            'title': self.titles[idx],
            'similarity': float(similarities[idx]),
            'id': self.ids[idx]
        })
```

```
    return {
        'input_title': new_title,
        'similar_titles': results
    }

def interpret_results(self, similar_pairs):
    """Interpret similarity detection results"""
    if not similar_pairs:
        self.log("No similar pairs detected")
        return None

    # Group similar pairs by similarity level
    high_similarity = [pair for pair in similar_pairs if pair['similarity'] >= 0.8]
    medium_similarity = [pair for pair in similar_pairs if 0.7 <= pair['similarity'] < 0.8]
    low_similarity = [pair for pair in similar_pairs if 0.6 <= pair['similarity'] < 0.7]

    report = {
        'total_pairs': len(similar_pairs),
        'high_similarity': len(high_similarity),
        'medium_similarity': len(medium_similarity),
        'low_similarity': len(low_similarity),
        'high_examples': sorted(high_similarity, key=lambda x: x['similarity'], reverse=True)[:5] if high_similarity else []
    }

    self.log("\nSimilarity Detection Report:")
    self.log(f"Total pairs with similarity above threshold: {report['total_pairs']}")
    self.log(f"High similarity pairs (≥ 0.8): {report['high_similarity']}")
```

```
    self.log(f"Medium similarity pairs (0.7-0.8): {report['medium_similarity']}")
    self.log(f"Low similarity pairs (0.6-0.7): {report['low_similarity']}")

    # Print some examples of highly similar pairs
    if high_similarity:
        self.log("\nTop 5 most similar pairs:")
        sorted_pairs = sorted(high_similarity, key=lambda x: x['similarity'],
reverse=True)
        for i, pair in enumerate(sorted_pairs[:5]):
            self.log(f"\nPair {i+1} (Similarity: {pair['similarity']:.4f}):")
            self.log(f"Title 1: {pair['title1']}")
            self.log(f"Title 2: {pair['title2']}")

    return report

def run_complete_pipeline(self):
    """Run the complete detection pipeline"""
    self.logs = []
    self.log("Starting thesis similarity detection pipeline...")

    # Step 1: Collect Data
    self.log("\n[Step 1] Collecting Data...")
    self.collect_data()

    # Step 2: Preprocess Data
    self.log("\n[Step 2] Preprocessing Data...")
    self.preprocess_data()

    # Step 3: Feature Extraction
```

```
self.log("\n[Step 3] Extracting Features (TF-IDF)...")
self.extract_features()

# Step 4: Train Model
self.log("\n[Step 4] Training Model...")
self.train_model()

# Step 5: Evaluate Model
self.log("\n[Step 5] Evaluating Model...")
self.evaluation_results = self.evaluate_model()

# Use the best threshold from evaluation
best_threshold = 0.6  # Default
if self.evaluation_results and 'best_threshold' in self.evaluation_results:
    best_threshold = self.evaluation_results['best_threshold']
    self.log(f"\nUsing best threshold from evaluation: {best_threshold}")

# Step 6: Detect Similarity
self.log(f"\n[Step 6] Detecting Similarity (threshold={best_threshold})...")
self.similar_pairs = self.detect_similarity(threshold=best_threshold)

# Step 7: Interpret Results
self.log("\n[Step 7] Interpreting Results...")
self.interpretation = self.interpret_results(self.similar_pairs)

self.log("\nThesis similarity detection pipeline completed!")
self.processed = True

# Return final results
```

```
    return {
        'evaluation': self.evaluation_results,
        'similar_pairs': self.similar_pairs,
        'interpretation': self.interpretation,
        'logs': self.logs
    }

def log(self, message):
    """Log a message"""
    timestamp = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
    log_entry = f"[{timestamp}] {message}"
    print(log_entry)
    self.logs.append(log_entry)
    return log_entry

def generate_metrics_chart(self):
    """Generate chart for metrics across different thresholds"""
    if not self.evaluation_results or 'all_thresholds' not in self.evaluation_results:
        return None

    thresholds = list(self.evaluation_results['all_thresholds'].keys())
    metrics = ['accuracy', 'precision', 'recall', 'f1_score']

    plt.figure(figsize=(10, 6))

    for metric in metrics:
        values = [self.evaluation_results['all_thresholds'][t][metric] for t in thresholds]
        plt.plot(thresholds, values, marker='o', label=metric.capitalize())
```

```
    plt.xlabel('Threshold')
    plt.ylabel('Score')
    plt.title('Evaluation Metrics by Threshold')
    plt.legend()
    plt.grid(True, linestyle='--', alpha=0.7)

    # Save plot to a temporary buffer
    buf = io.BytesIO()
    plt.savefig(buf, format='png')
    buf.seek(0)

    # Convert plot to base64 string
    img_str = base64.b64encode(buf.getvalue()).decode('utf-8')
    plt.close()

    return img_str

def generate_similarity_distribution_chart(self):
    """Generate distribution chart for similarity scores"""
    if not self.similar_pairs:
        return None

    similarities = [pair['similarity'] for pair in self.similar_pairs]

    plt.figure(figsize=(10, 6))
    sns.histplot(similarities, bins=20, kde=True)
    plt.xlabel('Similarity Score')
    plt.ylabel('Frequency')
    plt.title('Distribution of Similarity Scores')
```

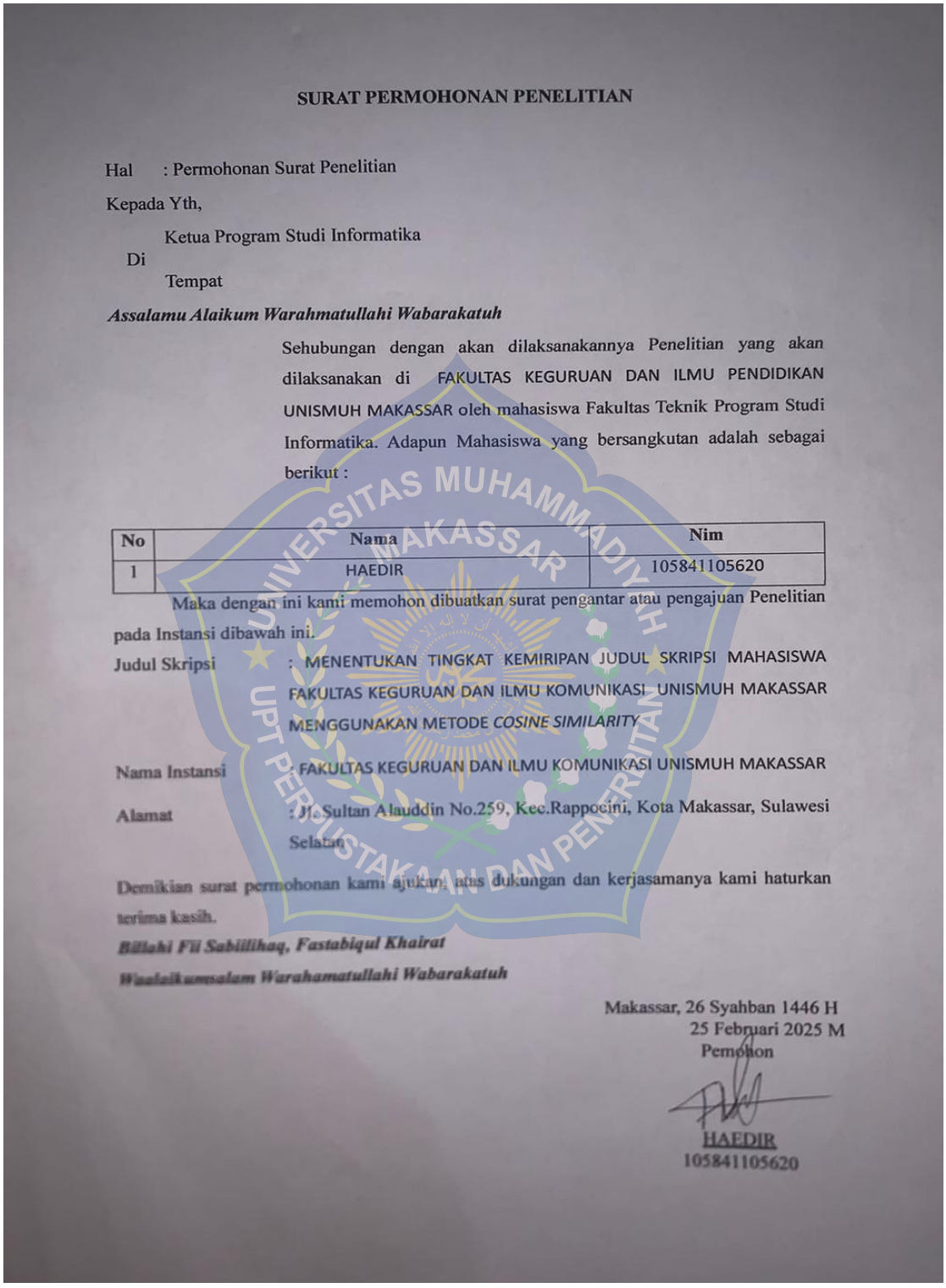
```
        # Save plot to a temporary buffer
        buf = io.BytesIO()
        plt.savefig(buf, format='png')
        buf.seek(0)

        # Convert plot to base64 string
        img_str = base64.b64encode(buf.getvalue()).decode('utf-8')
        plt.close()

        return img_str

# Initialize detector
detector = ThesisSimilarityDetector()

@app.route('/')
def index():
    return render_template('index.html', processed=detector.processed)

@app.route('/train', methods=['POST'])
def train():
    results = detector.run_complete_pipeline()
    metrics_chart = detector.generate_metrics_chart()
    similarity_chart = detector.generate_similarity_distribution_chart()

    return render_template(
        'results.html',
        results=results,
        processed=detector.processed,
```

```
        metrics_chart=metrics_chart,
        similarity_chart=similarity_chart
    )

@app.route('/check', methods=['GET', 'POST'])
def check_title():
    if request.method == 'POST':
        title = request.form.get('title', '')
        if not title:
            flash('Please enter a title to check')
            return redirect(url_for('check_title'))

        if not detector.processed:
            flash('Please train the model first')
            return redirect(url_for('index'))

        results = detector.check_title_similarity(title)
        return render_template('check_results.html', results=results,
        processed=detector.processed)

    return render_template('check.html', processed=detector.processed)

@app.route('/similar_pairs')
def view_similar_pairs():
    if not detector.processed:
        flash('Please train the model first')
        return redirect(url_for('index'))

    # Get filter parameters
```

```
min_similarity = request.args.get('min_similarity', 0.6, type=float)
max_similarity = request.args.get('max_similarity', 1.0, type=float)

# Filter pairs based on similarity range
filtered_pairs = [
    pair for pair in detector.similar_pairs
    if min_similarity <= pair['similarity'] <= max_similarity
]

# Sort by similarity (highest first)
filtered_pairs = sorted(filtered_pairs, key=lambda x: x['similarity'], reverse=True)

# Pagination
page = request.args.get('page', 1, type=int)
per_page = 10
total_pages = (len(filtered_pairs) + per_page - 1) // per_page

start_idx = (page - 1) * per_page
end_idx = min(start_idx + per_page, len(filtered_pairs))
current_pairs = filtered_pairs[start_idx:end_idx]

return render_template(
    'similar_pairs.html',
    pairs=current_pairs,
    page=page,
    total_pages=total_pages,
    min_similarity=min_similarity,
    max_similarity=max_similarity,
    processed=detector.processed
```

```
    )


@app.route('/logs')
def view_logs():
    return render_template('logs.html', logs=detector.logs,
        processed=detector.processed)


@app.route('/api/check_title', methods=['POST'])
def api_check_title():
    data = request.get_json()
    if not data or 'title' not in data:
        return jsonify({'error': 'No title provided'}), 400

    if not detector.processed:
        return jsonify({'error': 'Model not trained yet'}), 400

    results = detector.check_title_similarity(data['title'])
    return jsonify(results)

if __name__ == '__main__':
    app.run(debug=True)
```

Lampiran 2. Surat Permohonan Penelitian Kepada Ketua Program Studi Informatika



SURAT PERMOHONAN PENELITIAN

Hal : Permohonan Surat Penelitian

Kepada Yth,

Ketua Program Studi Informatika

Di

Tempat

Assalamu Alaikum Warahmatullahi Wabarakatuh

Sehubungan dengan akan dilaksanakannya Penelitian yang akan dilaksanakan di FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN UNISMUH MAKASSAR oleh mahasiswa Fakultas Teknik Program Studi Informatika. Adapun Mahasiswa yang bersangkutan adalah sebagai berikut :

No	Nama	Nim
1	HAEDIR	105841105620

Maka dengan ini kami memohon dibuatkan surat pengantar atau pengajuan Penelitian pada Instansi dibawah ini.

Judul Skripsi : MENENTUKAN TINGKAT KEMIRIPAN JUDUL SKRIPSI MAHASISWA FAKULTAS KEGURUAN DAN ILMU KOMUNIKASI UNISMUH MAKASSAR MENGGUNAKAN METODE *COSINE SIMILARITY*

Nama Instansi : FAKULTAS KEGURUAN DAN ILMU KOMUNIKASI UNISMUH MAKASSAR

Alamat : Jl. Sultan Alauddin No.259, Kec.Rappocini, Kota Makassar, Sulawesi Selatan

Demikian surat permohonan kami ajukan, atas dukungan dan kerjasamanya kami haturkan terima kasih.

Billahi Fii Sabilihaq, Fastabiqul Khairat

Waalaikumsalam Warahamatullahi Wabarakatuh

Makassar, 26 Syahban 1446 H
25 Februari 2025 M
Pemohon

HAEDIR
105841105620

Lampiran 3. Permohonan Penelitian Kepada LP3M Universitas Muhammadiyah Makassar

MAJELIS PENDIDIKAN TINGGI PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH MAKASSAR
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

Nomor : 016/05/IF/C.4-VI/III/46/2025
Lamp. : -
Hal : **Permohonan Data Penelitian**

Makassar, 11 Ramadhan 1446 H
11 Maret 2025 M

Kepada yang Terhormat,
Ketua LP3M Unismuh Makassar
Di –
Tempat

Assalamu 'Alaikum Warahmatullahi Wabarakatuh

Dengan Rahmat Allah SWT, Semoga aktivitas kita bernilai ibadah di Sisi–Nya. Dalam rangka penyelesaian Tugas Akhir pada Program Studi Informatika dengan judul "Menentukan Tingkat Kemiripan Judul Skripsi Mahasiswa Fakultas Keguruan dan Ilmu Pendididkan Unismuh Makassar Menggunakan Metode *Cosine Similarity*)" . Bersama ini kami sampaikan mahasiswa:

Stambuk	Nama
105 84 11056 20	Haedir

Sehubungan dengan hal tersebut, maka kami memohon dibuatkan surat pengantar pada instansi di bawah ini:

Nama Instansi : Fakultas Keguruan dan Ilmu Pendidikan Unismuh Makassar
Alamat : Jln. Sultan Alauddin No. 259, Kec. Rappocini, Kota Makassar, Sulawesi Selatan

Demikian surat kami atas perhatian dan kerja samanya kami haturkan banyak terima kasih.
Jazakumullah Khaeran Katsiran
Wassalamu 'Alaikum warahmatullah Wabarakatuh

Ketua Program Studi
Informatika,

Muhyiddin A M Hayat, S.Kom., M.T.
NBM. 1051187901

Tembusan:
1. Dekan Fakultas Teknik
2. Arsip

Lampiran 4. Surat Permohonan Izin Penelitian Kepada Dekan FKIP Unismuh Makassar

MAJELIS PENDIDIKAN TINGGI PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH MAKASSAR
LEMBAGA PENELITIAN PENGEMBANGAN DAN PENGABDIAN KEPADA MASYARAKAT
Jl. Sultan Alauddin No. 259 Telp.866972 Fax (0411)865588 Makassar 90221 e-mail :lp3m@unismuh.ac.id

Nomor : 6584/05/C.4-VIII/III/1446/2025
Lamp : 1 (satu) Rangkap Proposal
Hal : Permohonan Izin Penelitian

19 March 2025 M
19 Ramadhan 1446

Kepada Yth,
Dekan Fakultas Keguruan dan Ilmu Pendidikan
Universitas Muhamamdiyah Makassar
di –
Makassar

اَلسَّلَامُ عَلَيْكُمْ وَرَحْمَةُ اللهِ وَبَرَكَاتُهُ

Berdasarkan surat Dekan Fakultas Teknik Universitas Muhammadiyah Makassar, nomor: 016/05/IF/C.4-VI/III/46/2025 tanggal 11 Maret 2025, menerangkan bahwa mahasiswa tersebut di bawah ini :

Nama : **HAEDIR**
No. Stambuk : **10584 1105620**
Fakultas : **Fakultas Teknik**
Jurusan : **informatika**
Pekerjaan : **Mahasiswa**

Bermaksud melaksanakan penelitian/pengumpulan data dalam rangka penulisan Skripsi dengan judul :

"MENENTUKAN TINGKAT KEMIRIPAN JUDUL SKRIPSI MAHASISWA FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN UNISMUH MAKASSAR MENGGUNAKAN METODE COSINE SIMILARITY"

Yang akan dilaksanakan dari tanggal 21 Maret 2025 s/d 21 Mei 2025.

Sehubungan dengan maksud di atas, kiranya Mahasiswa tersebut diberikan izin untuk melakukan penelitian sesuai ketentuan yang berlaku.
Demikian, atas perhatian dan kerjasamanya diucapkan Jazakumullahu khaeran

اَلسَّلَامُ عَلَيْكُمْ وَرَحْمَةُ اللهِ وَبَرَكَاتُهُ

Ketua LP3M,

Dr. Muh. Ariel Muhsin, M.Pd.
NBM 1127761

Lampiran 5. Surat Keterangan Bebas Plagiat

MAJELIS PENDIDIKAN TINGGI PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH MAKASSAR
UPT PERPUSTAKAAN DAN PENERBITAN
Alamat kantor: Jl.Sultan Alauddin NO.259 Makassar 90221 Tlp.(0411) 866972,881593, Fax.(0411) 865588

بسم الله الرحمن الرحيم

SURAT KETERANGAN BEBAS PLAGIAT

UPT Perpustakaan dan Penerbitan Universitas Muhammadiyah Makassar,
Menerangkan bahwa mahasiswa yang tersebut namanya di bawah ini:,

Nama : Haedir
Nim : 105841105620
Program Studi : Teknik Informatika
Dengan nilai:

No	Bab	Nilai	Ambang Batas
1	Bab 1	8 %	10 %
2	Bab 2	19 %	25 %
3	Bab 3	9 %	15 %
4	Bab 4	5 %	10 %
5	Bab 5	3 %	5 %

Dinyatakan telah lulus cek plagiat yang diadakan oleh UPT- Perpustakaan dan Penerbitan Universitas Muhammadiyah Makassar Menggunakan Aplikasi Turnitin.

Demikian surat keterangan ini diberikan kepada yang bersangkutan untuk dipergunakan seperlunya.

Makassar, 20 Agustus 2025
Mengetahui
Kepala UPT- Perpustakaan dan Pernerbitan,

Nursinah, S.Hum.,M.I.P
NBM. 964591

Jl. Sultan Alauddin no 259 makassar 90222
Telepon (0411)866972,881 593,fax (0411)865 588
Website: www.library.unismuh.ac.id
E-mail : perpustakaan@unismuh.ac.id

BAB I Haedir 105841105620

by Tahap Tutup

Submission date: 15-Aug-2025 11:56AM (UTC+0700)
Submission ID: 2729860476
File name: BAB_1_45.docx (27.99K)
Word count: 750
Character count: 5097

BAB I Haedir 105841105620

ORIGINALITY REPORT

8%	6%	0%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

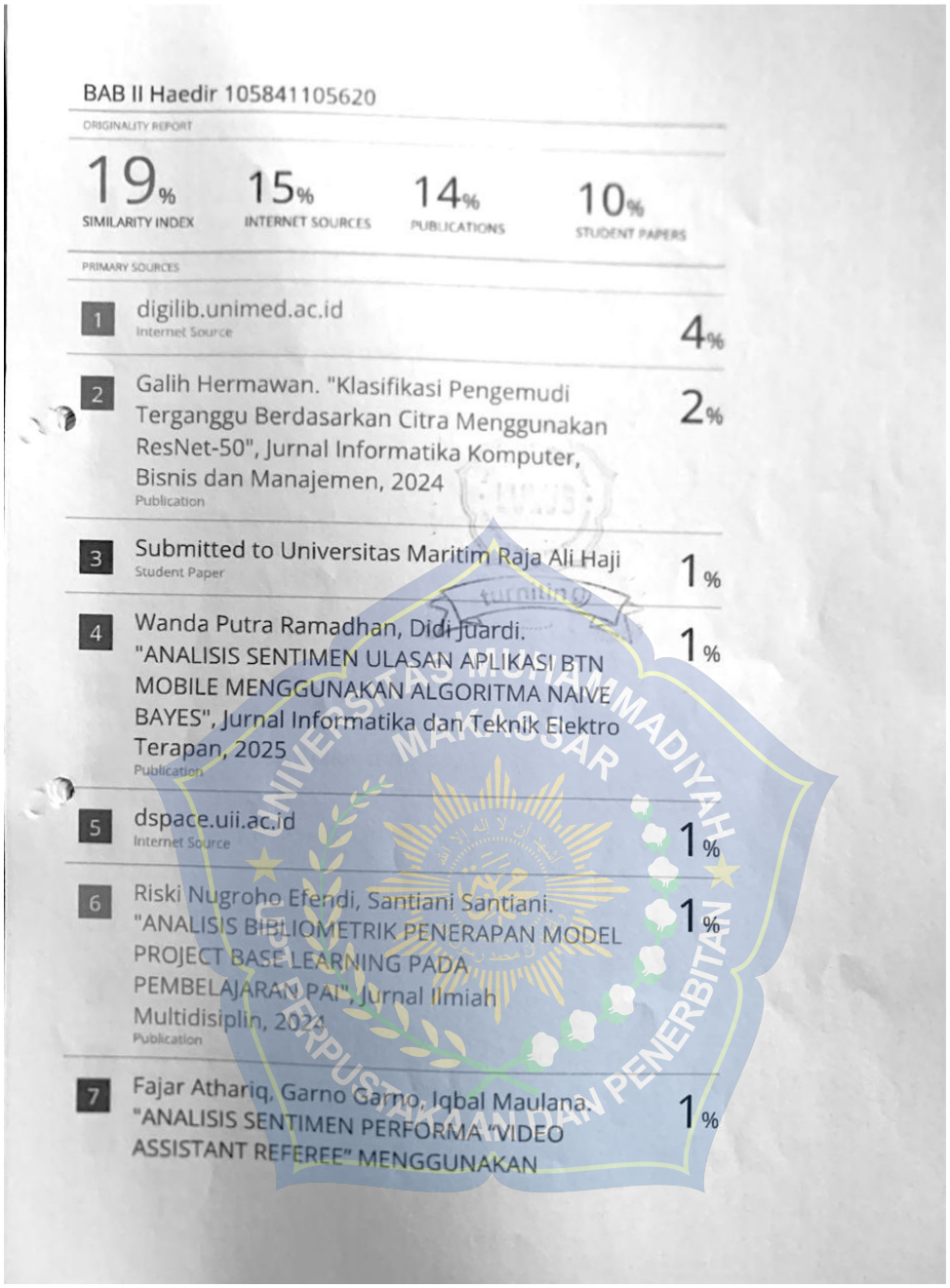
1	www.e-journal.stmiklombok.ac.id Internet Source	2%
2	Submitted to Universitas Airlangga Student Paper	2%
3	eprints.perbanas.ac.id Internet Source	2%
4	repo.itera.ac.id Internet Source	2%

Exclude quotes Off
Exclude bibliography Off
Exclude matches < 2%

BAB II Haedir 105841105620

by Tahap Tutup

Submission date: 15-Aug-2025 11:58AM (UTC+0700)
Submission ID: 2729861096
File name: BAB_2_46.docx (48.07K)
Word count: 1714
Character count: 11818



BAB II Haedir 105841105620

ORIGINALITY REPORT

19%	15%	14%	10%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	digilib.unimed.ac.id Internet Source	4%
2	Galih Hermawan. "Klasifikasi Pengemudi Terganggu Berdasarkan Citra Menggunakan ResNet-50", Jurnal Informatika Komputer, Bisnis dan Manajemen, 2024 Publication	2%
3	Submitted to Universitas Maritim Raja Ali Haji Student Paper	1%
4	Wanda Putra Ramadhan, Didi Juardi. "ANALISIS SENTIMEN ULASAN APLIKASI BTN MOBILE MENGGUNAKAN ALGORITMA NAIVE BAYES", Jurnal Informatika dan Teknik Elektro Terapan, 2025 Publication	1%
5	dspace.uii.ac.id Internet Source	1%
6	Riski Nugroho Efendi, Santiani Santiani. "ANALISIS BIBLIOMETRIK PENERAPAN MODEL PROJECT BASE LEARNING PADA PEMBELAJARAN PAI", Jurnal Ilmiah Multidisiplin, 2024 Publication	1%
7	Fajar Athariq, Garno Garno, Iqbal Maulana. "ANALISIS SENTIMEN PERFORMA "VIDEO ASSISTANT REFEREE" MENGGUNAKAN	1%

	ALGORITMA NAÏVE BAYES". Jurnal Informatika dan Teknik Elektro Terapan, 2024 Publication	
8	Fahri Patir Ramadhan, Muhammad Dzulfikar Apandi Putra, Muhammad Rafli Reinanda, Naufal Akmal Maulana, Zurnan Alfian. "Aplikasi Data Mining dalam Pengelompokan Informasi Multidisiplin", RIGGS: Journal of Artificial Intelligence and Digital Business, 2025 Publication	1 %
9	Vena Kurnia Sari, Dwi Hartanti, Eko Purwanto. "Sistem Rekomendasi Resep Masakan Menggunakan Metode Content Based Filtering Berdasarkan Preferensi Pengguna", Infotek: Jurnal Informatika dan Teknologi, 2025 Publication	1 %
10	jpti.journals.id Internet Source	1 %
11	yusrintosepu.wixsite.com Internet Source	1 %
12	repository.uin-suska.ac.id Internet Source	1 %
13	jurnal.ardenjaya.com Internet Source	1 %
14	ratu.ai Internet Source	1 %
15	repository.uph.edu Internet Source	1 %
16	lib.ui.ac.id Internet Source	<1 %

17	www.poloshirtsoutlet.us.com Internet Source	<1%
18	Arief Rahman Hakim, Alva Hendi Muhammad. "PERBANDINGAN MODEL TRANSFORMER, DEEP LEARNING, DAN MACHINE LEARNING UNTUK DETEKSI BERITA PALSU: STUDI KASUS PADA TEKS BERBAHASA INDONESIA", Jurnal Manajemen Informatika dan Sistem Informasi, 2025 Publication	<1%
19	Ferdy Febriyanto. "Sistem Penilaian Otomatis Jawaban Esai Dengan Menggunakan Metode Vector Space Model Pada Beberapa Perkuliahan Di Stmik Indonesia Banjarmasin", Respati, 2019 Publication	<1%
20	Jamilatun Safitri, Vihi Atina, Nugroho Arif Sudibyo. "Rancang bangun sistem rekomendasi pemilihan drama korea dengan metode content-based filtering", INFOTECH : Jurnal Informatika & Teknologi, 2024 Publication	<1%
21	ejurnal.itenas.ac.id Internet Source	<1%
22	jurnal.poliupg.ac.id Internet Source	<1%

Exclude quotes Off
Exclude bibliography Off
Exclude matches

BAB III Haedir 105841105620

by Tahap Tutup

Submission date: 15-Aug-2025 11:59AM (UTC+0700)
Submission ID: 2729861634
File name: BAB_3_43.docx (69.96K)
Word count: 876
Character count: 5802

BAB III Haedir 105841105620

ORIGINALITY REPORT

9%	6%	2%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

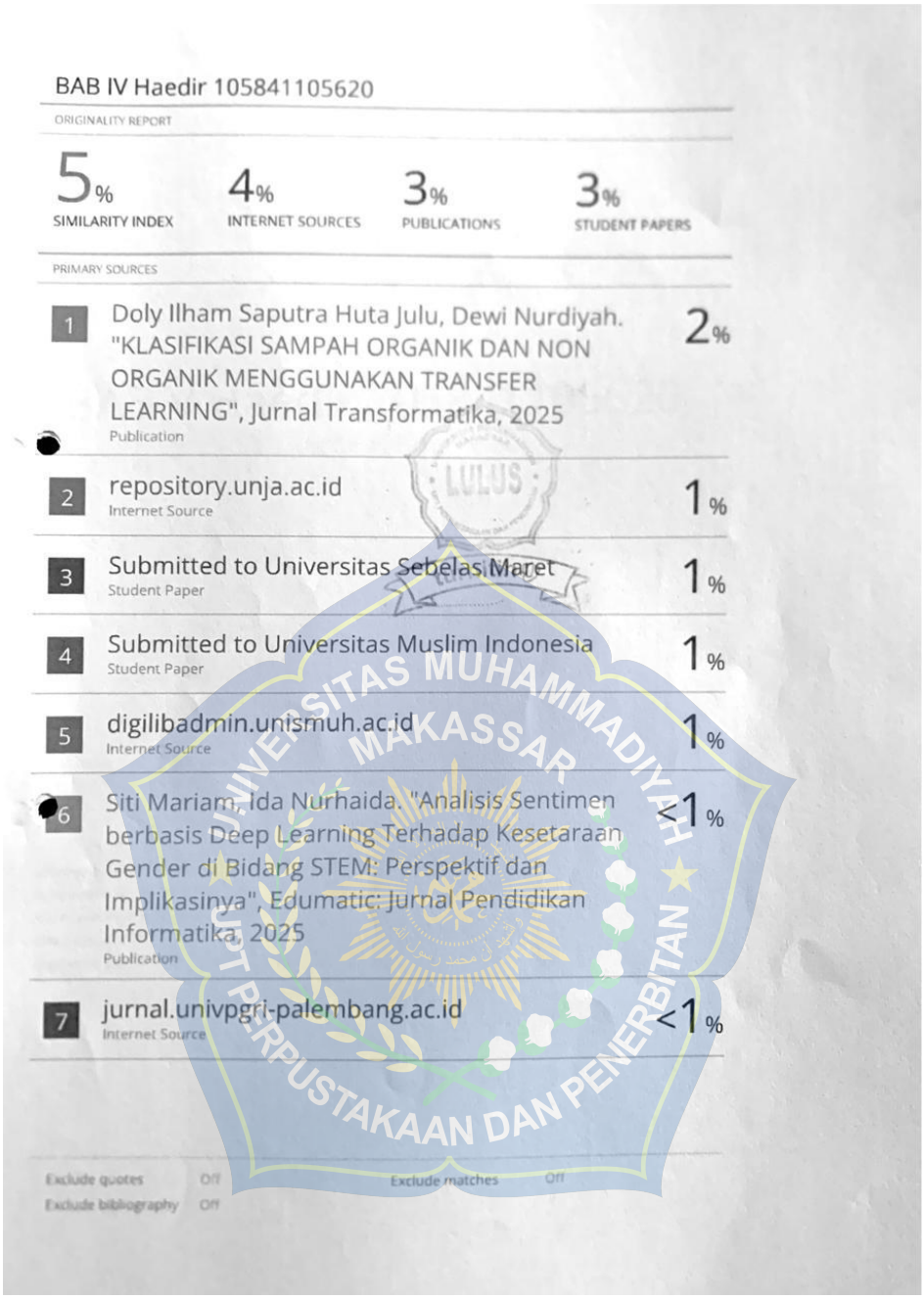
1	Submitted to Universitas Muhammadiyah Makassar Student Paper	2%
2	Submitted to Universitas 17 Agustus 1945 Surabaya Student Paper	2%
3	Nathanael Ferdian Putra Setyawan, Fauzan Nusyura, Ardian Yusuf Wicaksono, Farah Zakiyah Rahmanti. "Aplikasi Android untuk Rekomendasi Pemilihan Buah Anggur Hijau Menggunakan VGG16", Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi), 2024 Publication	2%
4	Submitted to itera Student Paper	2%
5	docplayer.info Internet Source	2%

Exclude quotes Off
Exclude bibliography Off
Exclude matches < 2%

BAB IV Haedir 105841105620

by Tahap Tutup

Submission date: 15-Aug-2025 12:00PM (UTC+0700)
Submission ID: 2729862298
File name: BAB_4_42.docx (741.52K)
Word count: 1677
Character count: 10942



BAB IV Haedir 105841105620

ORIGINALITY REPORT

5%	4%	3%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

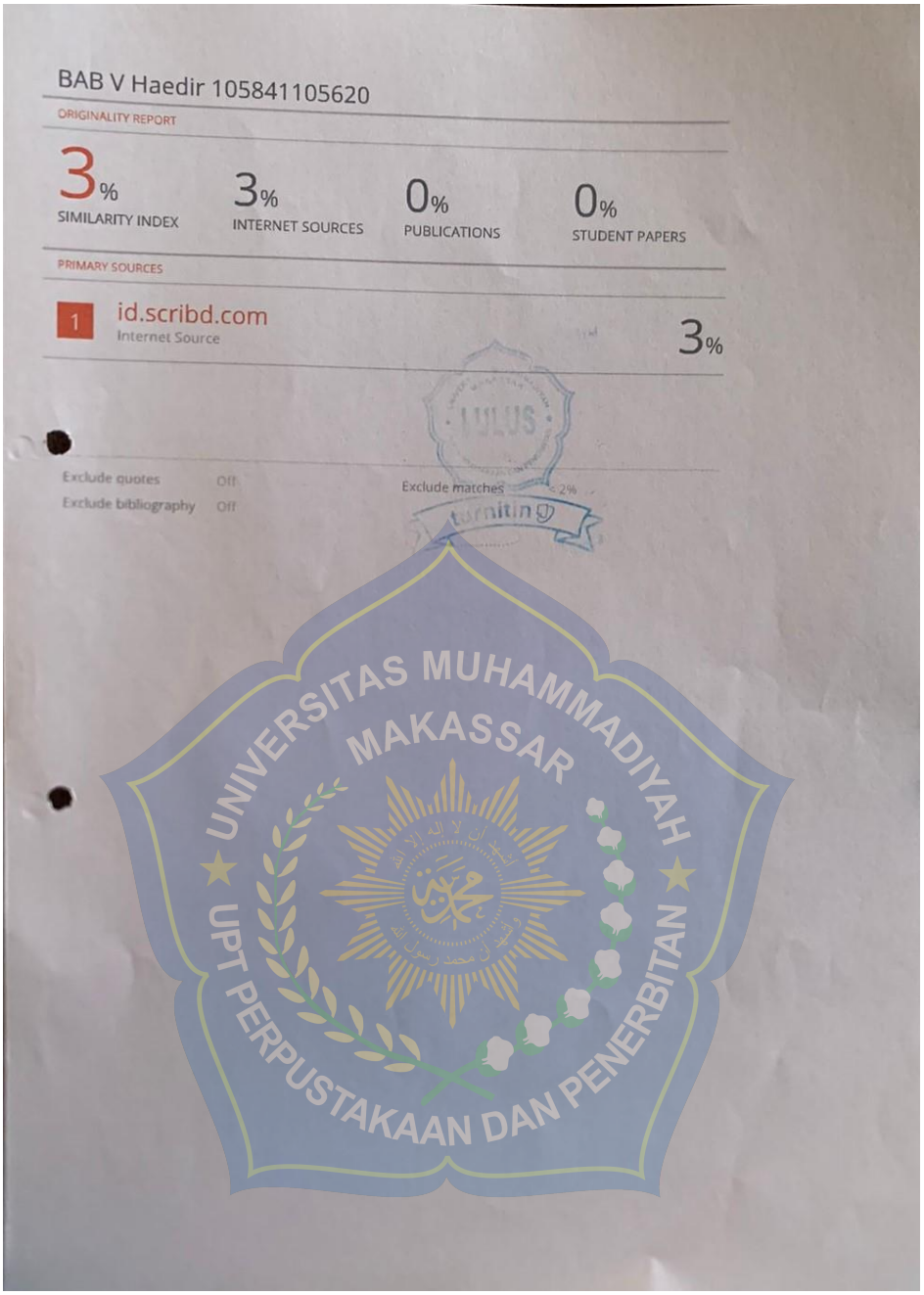
1	Doly Ilham Saputra Huta Julu, Dewi Nurdiyah. "KLASIFIKASI SAMPAH ORGANIK DAN NON ORGANIK MENGGUNAKAN TRANSFER LEARNING", Jurnal Transformatika, 2025 Publication	2%
2	repository.unja.ac.id Internet Source	1%
3	Submitted to Universitas Sebelas Maret Student Paper	1%
4	Submitted to Universitas Muslim Indonesia Student Paper	1%
5	digilibadmin.unismuh.ac.id Internet Source	1%
6	Siti Mariam, Ida Nurhaida. "Analisis Sentimen berbasis Deep Learning Terhadap Kesetaraan Gender di Bidang STEM: Perspektif dan Implikasinya", Edumatic: Jurnal Pendidikan Informatika, 2025 Publication	<1%
7	jurnal.univpgri-palembang.ac.id Internet Source	<1%

Exclude quotes Off
Exclude bibliography Off
Exclude matches Off


BAB V Haedir 105841105620

by Tahap Tutup

Submission date: 15-Aug-2025 12:01PM (UTC+0700)
Submission ID: 2729862720
File name: BAB_5_42.docx (21.73K)
Word count: 273
Character count: 1824



BAB V Haedir 105841105620

ORIGINALITY REPORT

3%	3%	0%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	id.scribd.com Internet Source	3%

LULUS

Exclude quotes Off
Exclude bibliography Off
Exclude matches < 2%

turnitin