IMPLEMENTASI *DEEP LEARNING* MENGGUNAKAN *HYBRID SENTENCE-TRANSFORMERS* DAN *K-MEANS* UNTUK PERBANDINGAN JURNAL

Muhammad Asygar Faeruddin*[1)], Muhammad Faisal [2)], Rizki Yusliana Bakti [3)], Muhammad Syafaat[4)], Muhyiddin AM Hayat[5)], Andi Makbul Syamsuri[6)], Andi Lukman Anas[7)]

1. Informatika, Universitas Muhammadiyah Makassar
asygar@student.unismuh.ac.id
2. Informatika, Universitas Muhammadiyah Makassar
muhfaisal@unismuh.ac.id
3. Informatika, Universitas Muhammadiyah Makassar
rizkiyusliana@unismuh.ac.id
4. Teknik Pengairan, Universitas Muhammadiyah Makassar
syafaat_skuba@unismuh.ac.id
5. Teknik Pengairan, Universitas Muhammadiyah Makassar
muhyiddin@unismuh.ac.id
6. Informatika, Universitas Muhammadiyah Makassar
amakbulsyamsuri@unismuh.ac.id
7. Informatika, Universitas Muhammadiyah Makassar
lukmananas@unismuh.ac.id

Abstract

This study addresses the challenge of identifying semantic relatedness between scientific journal articles by developing a classification system based on deep learning. The system applies an unsupervised learning approach using the Sentence-Transformers model and K-Means clustering to generate semantic similarity scores and categorical labels. Abstracts from journal PDFs are extracted and processed to determine similarity levels across four predefined categories. The optimal number of clusters was determined using Elbow Method, Silhouette Score, and Davies-Bouldin Index, resulting in k = 4. The system is implemented as a web-based application that allows users to upload two PDF files, compare them semantically, and receive both a similarity score and an AI-generated narrative explanation. Functional testing showed that all core features performed as expected. This system significantly reduces the time required to assess relatedness between journal articles, offering an efficient tool for academic research navigation.

***Kata Kunci**: Semantic Similarity, Sentence-Transformers, K-Means, Clustering, Deep Learning*

A. PENDAHULUAN

Pertumbuhan jumlah publikasi ilmiah di dunia mengalami peningkatan eksponensial, dengan rata-rata laju pertumbuhan sekitar 4,10% per tahun dan waktu penggandaan (*doubling time*) sebesar 17,3 tahun [1]. Lonjakan ini, didorong oleh kemajuan teknologi

informasi dan ketersediaan database digital, menimbulkan tantangan bagi peneliti dalam membandingkan, menelusuri, serta memahami keterkaitan antar jurnal yang relevan. Metode manual untuk menilai kemiripan antar dokumen seringkali memakan waktu lama dan rentan bias, sehingga diperlukan pendekatan otomatis yang mampu mengidentifikasi kemiripan semantik antar dokumen ilmiah secara efisien dan konsisten, sekaligus memfasilitasi perbandingan jurnal berdasarkan tingkat kemiripannya.

Teknologi pemrosesan bahasa alami (*Natural Language Processing*) kini banyak mengandalkan model berbasis *Transformer*, seperti BERT dan variannya, yang mampu menangkap hubungan kontekstual antar kata dengan lebih akurat. Berdasarkan arsitektur ini, *Sentence-Transformer* models dirancang khusus untuk tugas *semantic similarity*, menghasilkan dense *semantic embeddings* yang mempertahankan relasi maknawi teks dalam ruang vektor. Dengan representasi vektor ini, perhitungan kemiripan antar kalimat atau dokumen menggunakan metrik *cosine similarity* menjadi sangat andal, sehingga model ini sangat sesuai untuk tugas analisis kemiripan jurnal ilmiah berdasarkan kemiripan semantic [2].

Penggunaan *Sentence-Transformers* pada kombinasi dengan algoritma *K Means Clustering* telah terbukti efektif untuk mengelompokkan dokumen berdasarkan kedekatan semantik. Peneltian sebelumnya menunjukkan pengelompokkan opini masyarakat terhadap pelestarian ekosistem mangrove dan berhasil mengidentifikasi topik secara lebih kontekstual dibandingkan pendekatan LDA konvensional [3]. Hal ini menunjukkan bahwa *Sentence*-BERT mampu menangkap nuansa makna kalimat dan *K-Means* menjadi metode clustering yang efisien untuk dokumen tanpa label (*unsupervised*).

Penelitian terdahulu menunjukkan potensi dan batasan berbagai teknik *embedding* dan *clustering*. *Word2Vec* dan *K-Means* efektif untuk klasterisasi berita Indonesia, namun hanya pada level kata tanpa konteks kalimat penuh [4]. Pendekatan TF-IDF dengan *Cosine Similarity* pada data rekam medis, tetapi kurang optimal menangkap relasi semantik antar kalimat [5]. Perbandingan representasi teks menggunakan BERT dan TF-IDF dalam tugas *clustering*, dan menemukan bahwa BERT mengungguli TF-IDF pada 28 dari 36 metrik evaluasi [6]. Studi lainnya mengeksplorasi berbagai varian *Sentence*-BERT (SBERT) dan teknik *pooling*, dan menemukan bahwa *mean pooling* memberikan hasil *clustering* terbaik di sebagian besar tugas, menegaskan fleksibilitas SBERT dalam skenario *transfer learning* [7]. Prinsip *embedding* berbasis Transformer juga diperkuat oleh pemanfaatan embedding semantik GPT-3 dan algoritma HDBSCAN dalam identifikasi topik utama pada publikasi ilmiah [8]. Oleh karena itu, penelitian ini mengusulkan penggunaan model *embedding* kontekstual berbasis *Sentence-Transformers* (SBERT) yang dikombinasikan dengan algoritma *K-Means*, dengan tujuan meningkatkan akurasi perbandingan kemiripan semantik antar jurnal ilmiah.

Meskipun beberapa penelitian sebelumnya telah menerapkan pendekatan semantik pada kumpulan teks, namun fokus penelitian tersebut masih terbatas di berbagai domain tertentu seperti media sosial, berita, dan data medis. Sedangkan pada proses analisis kemiripan semantik antar jurnal, masih terdapat keterbatasan ketersediaan dataset berlabel. Hal ini menyulitkan penerapan metode *supervised* yang memerlukan pelabelan manual dan *fine-tuning* model. Oleh

karena itu, pendekatan tanpa pengawasan (*unsupervised*) seperti kombinasi *Sentence-Transformers* dan *K-Means* menjadi solusi yang relevan untuk peneltian ini.

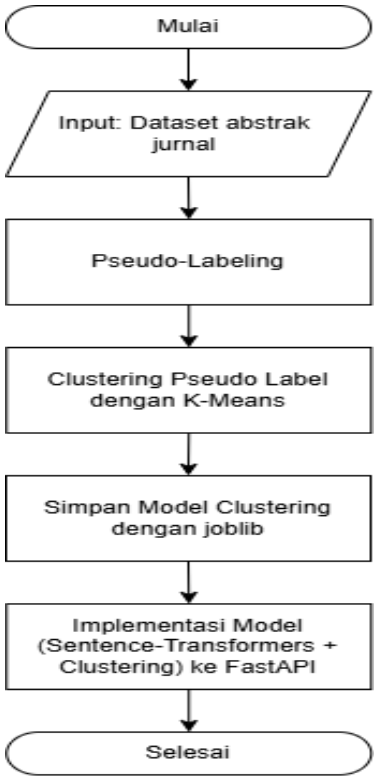
Berdasarkan permasalahan yang ditemukan, penelitian ini bertujuan menerapkan kombinasi *Sentence Transformers* dan *K-Means* untuk mengelompokkan jurnal ilmiah berdasarkan kemiripan semantik abstraknya. Dalam rangka penerapan teori NLP dan *Machine Learning*, untuk membangun sistem otomatis yang dapat mempermudah analisis komparatif jurnal secara kontekstual.

B. METODE PENELITIAN

Untuk merealisasikan tujuan penelitian, pendekatan dan metode yang diterapkan disusun secara sistematis dan berbasis teori yang telah terbukti efektivitasnya dalam konteks serupa. Bagian ini menjelaskan rancangan sistem, teknik pengumpulan dan analisis data, serta justifikasi pemilihan metode dan algoritma yang digunakan.

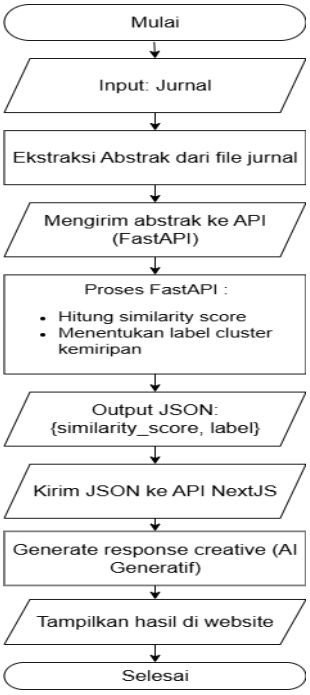
1) Penelitian ini mengadopsi pendekatan *unsupervised learning* untuk mengelompokkan jurnal ilmiah berdasarkan kemiripan semantik antar abstrak, tanpa memerlukan label manual. Pendekatan ini didasarkan pada kemampuan sistem dalam mengenali pola dan struktur laten dalam data tidak berlabel [9]. Salah satu teknologi utama yang digunakan adalah model *Sentence-Transformers* (SBERT), yang merupakan pengembangan dari arsitektur BERT dengan penambahan *pooling layer* untuk menghasilkan representasi vektor kalimat berdimensi tetap [10]. Representasi ini dikenal sebagai *sentence embeddings*, dan telah terbukti efektif dalam berbagai tugas seperti klasifikasi teks, deteksi rumor, dan pengukuran kemiripan semantik antar dokumen [10][11]. Untuk mengukur tingkat kemiripan antar abstrak jurnal, sistem ini menggunakan pendekatan *semantic similarity*, yang menghitung kesamaan makna menggunakan representasi vektor, bukan sekadar kemiripan kata kunci [12]. Nilai kemiripan ini kemudian digunakan dalam proses *pseudo-labeling*, sebuah teknik *semi-supervised* yang memungkinkan pemanfaatan data tidak berlabel untuk pelatihan model dengan efisiensi tinggi [13]. Dalam penelitian ini, *pseudo-labeling* berbasis nilai kemiripan digunakan sebagai dasar klasterisasi menggunakan algoritma *K-Means*, yang terkenal karena efisiensinya dalam mengelompokkan data berdasarkan kedekatan antar vektor [14]. Kombinasi SBERT, *semantic similarity*, *pseudo-labeling*, dan *K-Means* menjadikan pendekatan ini cocok untuk tugas komparasi jurnal yang menuntut objektivitas, kecepatan, dan skalabilitas.
2) Proses pengumpulan data dilakukan secara daring melalui Application Programming Interface (API) dari *Semantic Scholar*, yang menyediakan data jurnal ilmiah multidisiplin untuk keperluan ekstraksi metadata dan konten abstrak.
3) Tahapan awal dari sistem yang dikembangkan dimulai dengan proses pembentukan model yang terdiri dari dua komponen utama, yaitu perhitungan skor kemiripan semantik menggunakan *Sentence-Transformers* dan proses klasterisasi menggunakan algoritma *K-Means*. Model embedding yang digunakan menghasilkan representasi vektor dari setiap abstrak jurnal, kemudian dilakukan perhitungan *cosine similarity* antar pasangan abstrak untuk memperoleh nilai kemiripan.

Nilai-nilai ini selanjutnya dijadikan *pseudo-label* yang menjadi dasar proses klasterisasi. Model klaster hasil pelatihan kemudian disimpan dalam bentuk serialisasi untuk digunakan pada sistem produksi. Proses ini divisualisasikan pada Gambar 1.



Gambar 1. Flowchart Perancangan Model dan Clustering

Setelah tahap pelatihan model selesai, sistem dilanjutkan pada tahap penerapan dalam bentuk aplikasi berbasis *Application Programming Interface* (API). Pada tahap ini, sistem menerima input berupa dua file jurnal dalam format PDF, mengekstrak bagian abstrak dari masing-masing file, lalu mengirimkan data ke server *backend* untuk diproses. *Backend* menghitung skor kemiripan menggunakan model *Sentence-Transformers*, dan menetapkan label klaster kemiripan berdasarkan model *K-Means*. Hasil akhir berupa skor kemiripan, label kategorikal, dan narasi perbandingan dalam dua bahasa ditampilkan kepada pengguna melalui antarmuka web. Alur kerja keseluruhan proses ini digambarkan dalam Gambar 2.



Gambar 2. Flowchart Pengolahan Jurnal dan Output

4) Setelah proses ekstraksi teks dari file jurnal, sistem melakukan pra-pemrosesan awal terhadap bagian abstrak yang berhasil diambil. Tahapan ini meliputi pembersihan karakter non-alfabet, normalisasi teks, serta penghapusan whitespace berlebih. Proses ini penting untuk memastikan bahwa input yang diteruskan ke model embedding bersifat bersih dan konsisten. Selanjutnya, setiap abstrak diubah menjadi representasi numerik berdimensi tetap menggunakan model *Sentence-Transformers*, yang dirancang khusus untuk menangkap hubungan semantik antar kalimat. Proses ini menghasilkan *dense semantic embeddings* yang dapat digunakan untuk perhitungan kemiripan antar dokumen secara

matematis di ruang vektor berdimensi tinggi. Representasi ini menjadi dasar bagi seluruh proses analitik dalam sistem.

5) Setelah memperoleh embedding dari masing-masing abstrak jurnal, tahap selanjutnya adalah menghitung tingkat kemiripan semantik antar pasangan dokumen. Metode yang digunakan dalam penelitian ini adalah *cosine similarity*, yang secara matematis mengukur sudut antara dua vektor dalam ruang berdimensi tinggi. Pendekatan ini sangat umum digunakan dalam bidang pemrosesan bahasa alami karena mampu mencerminkan kedekatan makna antar representasi teks yang telah dienkode dalam bentuk vektor. Rumus perhitungan *cosine similarity* antara dua *vector* A dan B ditunjukkan pada Persamaan (1):

$$cosSim(A,B) = \frac{A \times B}{\|A\| \times \|B\|} \quad (1)$$

Nilai *cosine similarity* berada pada rentang [−1,1][-1, 1][−1,1], di mana nilai mendekati 1 menunjukkan tingkat kemiripan semantik yang tinggi antara dua teks. Penggunaan metrik ini telah terbukti efektif dalam berbagai studi untuk mendeteksi kemiripan antar kalimat maupun dokumen [15].

6) Setelah skor kemiripan antar pasangan abstrak dihitung menggunakan *cosine similarity*, sistem melakukan proses *pseudo-labeling* dengan menjadikan skor tersebut sebagai dasar pelabelan otomatis terhadap pasangan dokumen. Proses ini tidak melibatkan anotasi manual, melainkan sepenuhnya bergantung pada nilai similarity sebagai proksi label tingkat kemiripan. Strategi ini termasuk dalam pendekatan *self-training* dalam pembelajaran *semi-supervised*, di mana model memanfaatkan data tidak berlabel untuk menghasilkan anotasi semu (*pseudo-labels*) sebagai data latih tambahan. Pendekatan ini banyak digunakan untuk menghemat biaya pelabelan dan menjaga konsistensi evaluasi, serta telah terbukti efektif dalam tugas klasifikasi berbasis teks berskala besar [13].

7) Untuk mengelompokkan pasangan jurnal berdasarkan tingkat kemiripan yang diperoleh dari pseudo-label, penelitian ini menggunakan algoritma K-Means Clustering. K-Means merupakan metode *partitional clustering* yang membagi data ke dalam sejumlah klaster berdasarkan kedekatan antar data dalam ruang vektor. Algoritma ini bekerja dengan menetapkan sejumlah klaster awal, lalu mengelompokkan data berdasarkan jarak terdekat ke masing-masing centroid, kemudian memperbarui posisi centroid hingga mencapai konvergensi. Jarak antar data dan centroid dihitung menggunakan Euclidean Distance, sebagaimana dirumuskan pada Persamaan (2):

$$d(O_{i,}O_j) = \sqrt{\sum_{d=1}^{p}(O_{id} - O_{jd})^2} \quad (2)$$

Di mana O_i dan O_j menyatakan dua vektor data, p adalah jumlah dimensi, dan O_{id} adalah nilai fitur ke-d dari vektor O_i. K-Means dipilih karena efisien, sederhana, dan cukup efektif untuk mengelompokkan data *embedding* berdimensi tinggi, seperti yang digunakan dalam studi serupa untuk pengelompokan teks [16].

C. HASIL DAN PEMBAHASAN

1. Pengumpulan Dataset

Dataset yang digunakan dalam penelitian ini diperoleh melalui pemanfaatan *Application Programming Interface* (API) dari *Semantic Scholar*, yang menyediakan akses terhadap koleksi jurnal ilmiah lintas disiplin secara daring. Dari proses pengambilan dan seleksi data, diperoleh sebanyak 500 entri jurnal yang masing-masing memiliki *metadata* lengkap, termasuk judul, nama penulis, dan abstrak. Fokus utama dalam penelitian ini adalah pada isi abstrak dari setiap jurnal sebagai representasi semantik yang dianalisis. Dataset kemudian dibentuk menjadi pasangan jurnal dalam format tabel, di mana setiap baris terdiri atas empat atribut utama: *title1*, *abstract1*, *title2*, *abstract2*. Struktur ini memungkinkan sistem untuk melakukan komparasi antar abstrak jurnal secara sistematis pada tahap-tahap selanjutnya dalam penelitian.

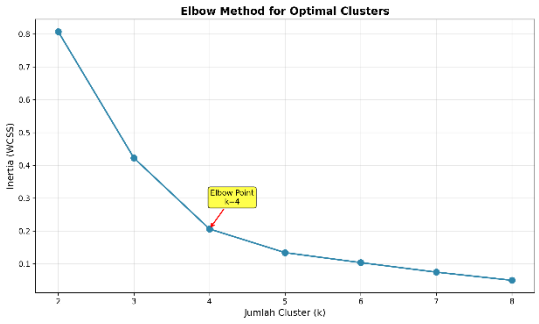
2. Pseudo-Labeling

Tahapan selanjutnya setelah pembentukan dataset adalah melakukan proses *pseudo-labeling* untuk setiap pasangan jurnal dengan menggunakan model *Sentence-Transformers* varian *all-mpnet-base-v2*. Model ini merupakan salah satu arsitektur SBERT yang dirancang untuk menghasilkan representasi kalimat berbasis konteks dalam bentuk vektor berdimensi tetap. Masing-masing abstrak dalam satu pasangan diproses oleh model untuk menghasilkan vektor *embedding*, lalu dibandingkan menggunakan perhitungan *cosine similarity* guna memperoleh skor kemiripan semantik. Nilai *similarity* yang dihasilkan berada dalam rentang 0 hingga 1, di mana skor mendekati 1 menunjukkan bahwa kedua abstrak memiliki kesamaan makna yang tinggi. Berdasarkan nilai ini, sistem menetapkan *pseudo-label* secara otomatis tanpa pelabelan manual, dengan mengelompokkan skor ke dalam beberapa kategori tingkat kemiripan. Dataset hasil *pseudo-labeling* ini kemudian menjadi dasar dalam proses klasterisasi untuk mengelompokkan jurnal secara semantik.

3. Evaluasi Klaster Optimal

Setelah proses *pseudo-labeling* selesai, langkah berikutnya adalah mengevaluasi jumlah klaster optimal yang akan digunakan dalam proses pengelompokan jurnal. Evaluasi dilakukan menggunakan tiga metrik yang umum dipakai dalam tugas *unsupervised clustering*, yaitu *Elbow Method*, *Silhouette Score*, dan *Davies-Bouldin Index* (DBI). Masing-masing metrik memberikan perspektif berbeda dalam menilai kualitas klaster yang terbentuk.

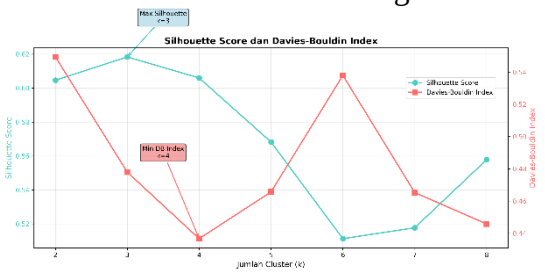
Visualisasi pertama ditunjukkan pada Gambar 3, yaitu *Elbow Method* yang menggambarkan hubungan antara jumlah klaster dan nilai *within-cluster sum of squares (WCSS)*. Pada grafik ini, terlihat adanya titik tekuk atau *elbow* yang muncul pada nilai k=4, yang mengindikasikan bahwa setelah titik tersebut, penambahan klaster tidak lagi memberikan penurunan t signifikan pada nilai WCSS.



Gambar 3. Grafik *Elbow Method* terhadap Variasi Nilai *k*

Gambar 4 memperlihatkan dua metrik lain, yaitu *Silhouette Score* dan *Davies-Bouldin Index* (DBI). Nilai *Silhouette* mencapai puncaknya pada *k*=3, menandakan bahwa pembagian data ke dalam tiga klaster memberikan pemisahan

yang paling baik antar kelompok. Sementara itu, nilai DBI yang semakin kecil pada *k*=4 menunjukkan bahwa jarak antar klaster semakin optimal dan struktur klaster semakin terdefinisi dengan baik.



Gambar 4. Grafik *Silhouette Score* (Biru) dan DBI (Pink) terhadap Nilai *k*

Untuk merangkum hasil dari ketiga metrik evaluasi tersebut, disajikan Tabel 1 yang memperlihatkan nilai kuantitatif dari masing-masing metrik terhadap variasi jumlah klaster yang diuji.

Tabel 1. Hasil Evaluasi Klaster Menggunakan *Elbow, Silhouette*, dan DBI

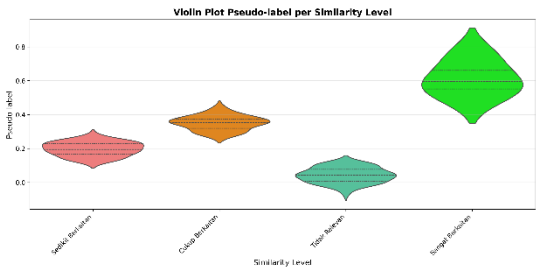
k	*Elbow Method*	*Silhoutte Score*	*Davies-Bouldin Index* (DBI)
2	Penurunan tajam	Skor tinggi, pemisahan kasar	DBI tinggi, klaster belum optimal
3	Penurunan berlanjut	Skor tertinggi	DBI render
4	Melandai	Skor stabil tinggi	DBI terendah
5	Penurunan minim	Skor menurun	DBI mulai naik
6	Hampir datar	Skor lebih rendah	DBI naik drastis
7	Hampir datar	Skor sedikit naik	DBI tetap tinggi
8	Hampir datar	Skor sedikit naik	DBI naik kembali

Tabel 1 merangkum nilai numerik dari masing-masing metrik evaluasi terhadap beberapa variasi jumlah klaster. Dari ketiga metrik yang diuji, semuanya mengarah pada kesimpulan bahwa *k*=4 adalah nilai klaster paling optimal dalam penelitian ini.

4. Proses Klastering *Pseudo-Label*

Pada tahap ini bertujuan untuk mengelompokkan data hasil *pseudo-labeling* ke dalam sejumlah klaster yang merepresentasikan tingkat kemiripan semantik antar jurnal. Proses pengelompokan dilakukan menggunakan algoritma *K-Means Clustering*, berdasarkan jumlah klaster optimal yang telah ditentukan melalui evaluasi sebelumnya menggunakan metrik *Davies-Bouldin Index* (DBI), *Elbow Method*, dan *Silhouette Score*. Adapun label kategorikal yang merefleksikan derajat kemiripan, yaitu tidak relevan, sedikit berkaitan, cukup berkaitan dan sangat berkaitan.

Pemberian label ini bertujuan agar setiap kelompok dapat diinterpretasikan secara semantik dan dimanfaatkan dalam analisis maupun proses naratif yang mendalam pada tahapan sistem lanjutan. Gambar 5 berikut menunjukkan hasil visualisasi distribusi *pseudo-label* terhadap empat kategori kemiripan:



Gambar 5. Visualisasi klastering *pseudo-label* pada ketegori

Berdasarkan Visualisasi Violin Plot pada Gambar 10, distribusi setiap kategori menunjukkan ciri khas yang konsisten dengan tingkat kemiripannya:

- Tidak Relevan (warna hijau toska): menyebar pada rentang nilai rendah (di bawah 0.12), dengan rata-rata sekitar 0.042.
- Sedikit Berkaitan (warna merah muda): memiliki nilai yang lebih tinggi, berkisar antara 0.13 hingga 0.27, dengan rata-rata sekitar 0.197.

- Cukup Berkaitan (warna oranye): nilai *pseudo-label* lebih terkonsentrasi pada rentang menengah (0.28–0.44), dengan rata-rata 0.351.
- Sangat Berkaitan (warna hijau terang): distribusi terletak pada rentang tertinggi, mulai dari 0.51 hingga 0.75, dengan rata-rata sebesar 0.615.

Tabel 2 merangkum statistik deskriptif dari distribusi nilai *pseudo-label* pada masing-masing kategori:

Tabel 2. Statistik Deskriptif *Pseudo-Label* per Klaster

Level	Min	Max	Mean	Median	Std
Tidak Relevan	-0.066	0.118	0.042	0.043	0.045
Sedikit Berkaitan	0.126	0.273	0.197	0.196	0.041
Cukup Berkaitan	0.279	0.435	0.351	0.354	0.042
Sangat Berkaitan	0.509	0.754	0.615	0.598	0.105

Distribusi *pseudo-label* pada tiap klaster menunjukkan pola konsisten pada tingkat kemiripan semantik antar jurnal. Setiap kategori memiliki rentang nilai yang relatif terpisah dan rata-rata skor yang meningkat dari kategori Tidak Relevan hingga Sangat Berkaitan. Ini mencerminkan keberhasilan *pseudo-labeling* dan klasterisasi dalam mengelompokkan pasangan jurnal secara logis dan terstruktur. Selanjutnya, Model *K-Means* disimpan dalam format JOBLIB, sedangkan pemetaan indeks klaster ke label kategorikal disimpan dalam bentuk JSON. Langkah ini memastikan model dan label siap digunakan pada tahap implementasi sistem.

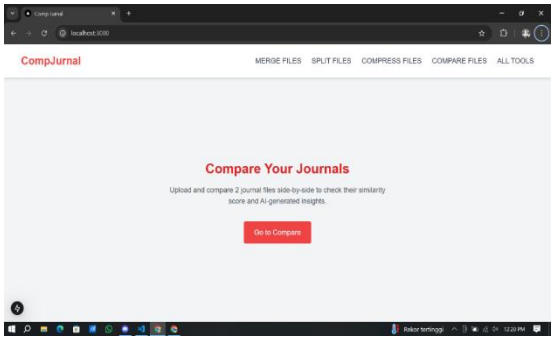
5. Implementasi Sistem

Tahap selanjutnya dalam penelitian ini adalah implementasi sistem berbasis web yang mengintegrasikan model pembelajaran mesin dan antarmuka pengguna (*user interface*) untuk melakukan perbandingan jurnal secara otomatis. Sistem ini dibangun menggunakan arsitektur dua lapis, yaitu *backend Python* berbasis *FastAPI* dan *frontend-backend* Next.js yang terintegrasi dengan layanan AI untuk pembuatan ringkasan.

Adapun alur sistem dapat dijelaskan melalui tahapan implementasi sebagai berikut:

- Halaman Awal Sistem (*HomePage*)

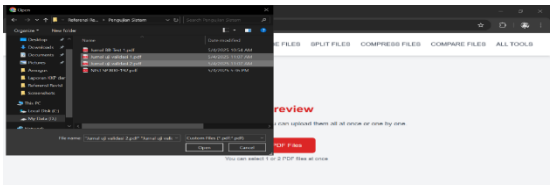
Gambar 6 berikut menunjukkan tampilan awal antarmuka sistem ketika pengguna mengakses halaman utama. Tombol "*Go to Compare*" akan mengarahkan pengguna untuk memulai proses perbandingan jurnal.



Gambar 6. Tampilan halaman utama sistem (*home page*)

- Unggah file

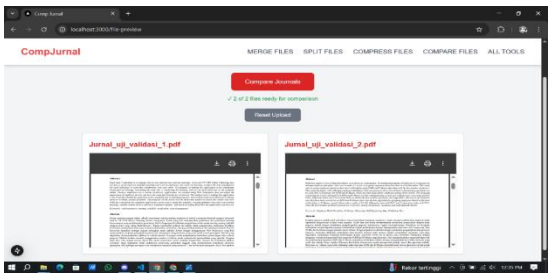
Pengguna diminta untuk mengunggah dua file jurnal dalam format PDF. Sistem hanya mengizinkan tepat dua file untuk dianalisis. Tampilan ini ditunjukkan pada Gambar 7.



Gambar 7. Proses unggah dua jurnal

- Pratinjau File Jurnal

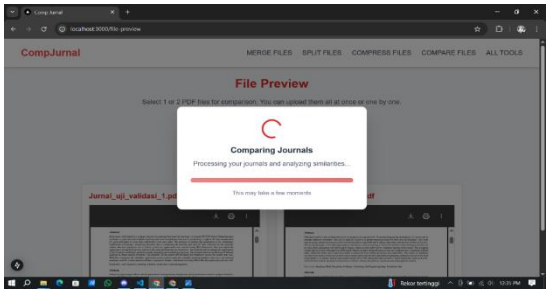
Setelah file berhasil diunggah, sistem akan menampilkan pratinjau kedua jurnal untuk memastikan isi dokumen sebelum dilakukan analisis. Ilustrasi tampilan ini dapat dilihat pada Gambar 8:



Gambar 8. Pratinjau file jurnal sebelum proses komparasi

- Proses Komparasi

Setelah pengguna menekan tombol "*Compare*", sistem akan mengekstrak abstrak dari kedua jurnal, menghitung skor kemiripan menggunakan *backend* Python, dan mengklasifikasikan hasilnya ke dalam label kategorikal. Gambar 9 menunjukkan proses *loading* saat sistem menjalankan komputasi:



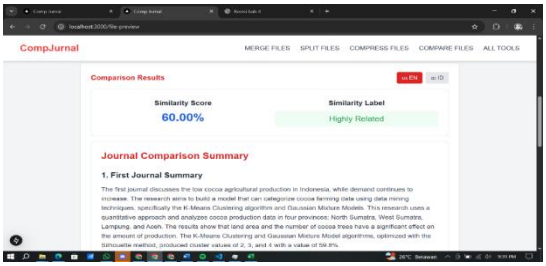
Gambar 9. Proses komparasi sistem dalam melakukan perbandingan

- Tampilan hasil komparasi

Hasil perbandingan ditampilkan terdiri dari:

1. Skor Kemiripan (%)
2. Label Kemiripan Kategorikal
3. Ringkasan AI dalam enam bagian utama:
    - Ringkasan Jurnal Pertama
    - Ringkasan Jurnal Kedua
    - Kesamaan Utama
    - Perbedaan Utama
    - Analisis Hubungan
    - Kesimpulan

Gambar 10 menampilkan hasil perbandingan



Gambar 10. Hasil komparasi dan ringkasan jurnal

Dengan adanya implementasi sistem ini, pengguna dapat:

- Melakukan perbandingan jurnal secara otomatis dan efisien.
- Mendapatkan interpretasi semantik dari kesamaan jurnal.
- Melihat hasil analisis dari perbandingan 2 jurnal.
- Mengidentifikasi keterkaitan, perbedaan, dan relevansi antar dua karya ilmiah.

6. Pengujian Sistem

Pengujian sistem bertujuan untuk memastikan bahwa fungsionalitas sistem telah berjalan sesuai dengan tujuan perancangan serta mampu menangani skenario penggunaan secara utuh. Pada penelitian ini, dilakukan pengujian dengan menggunakan metode *Blackbox Testing*, yaitu pendekatan pengujian perangkat lunak yang berfokus pada aspek fungsional sistem tanpa memeriksa struktur internal atau kode program.

Tabel 3 berikut menyajikan skenario pengujian sistem berdasarkan *test case* yang telah dirancang sesuai alur utama aplikasi:

Tabel 3. Hasil Pengujian Sistem dengan Metode *Blackbox Testing*

Test Case	Skenario Pengujian	Hasil yang di Harapkan	Hasil
Unggah Dua Jurnal PDF	Pengguna mengklik tombol "*Go to Compare*" dari halaman beranda	Sistem dapat menampilkan pratinjau isi kedua jurnal	Berhasil
Validasi unggahan dua jurnal	Pengguna mencoba mengunggah lebih dari dua jurnal	Sistem menolak unggahan dan menampilkan pesan kesalahan	Berhasil
Respons sistem terhadap teks kosong	Pengguna mengunggah jurnal dengan abstrak kosong atau rusak	Sistem memberikan notifikasi kesalahan input	Berhasil
Proses Komparasi Jurnal	Sistem memproses kemiripan semantik berdasarkan abstrak dari dua jurnal yang telah diunggah	Sistem menampilkan skor kemiripan dalam persentase serta label kemiripan kategori	Berhasil
Hasil Skor Kemiripan Rendah	Sistem diuji dengan dua jurnal berbeda topik: kanker dan peramalan air	Sistem menampilkan skor kemiripan rendah (≈10%) dan label "Tidak Relevan"	Berhasil
Hasil Skor Kemiripan Menengah	Sistem diuji dengan jurnal terkait sistem rekomendasi dan *black-box testing*	Sistem menampilkan skor sekitar 19% dan label "Sedikit Berkaitan"	Berhasil
Hasil Skor Kemiripan Cukup Terkait	Sistem diuji dengan dua jurnal yang membahas model embedding NLP	Sistem menampilkan skor sekitar 30% dan label "Cukup Berkaitan"	Berhasil
Hasil Skor Kemiripan Tinggi	Sistem diuji dengan dua jurnal yang membahas pengelompokan dokumen menggunakan BERT	Sistem menampilkan skor tinggi (≈75%) dan label "Sangat Berkaitan"	Berhasil
Tampilan Bahasa AI *Narrative* (EN & ID)	Pengguna memilih mode tampilan narasi perbandingan hasil dalam bahasa Inggris dan Bahasa Indonesia	Sistem menampilkan narasi perbandingan AI dalam dua bahasa (EN & ID) berdasarkan hasil analisis kedua jurnal	Berhasil
Respons Sistem terhadap *Reset Upload*	Pengguna mengklik tombol "*Reset Upload*" setelah mengunggah dua jurnal	Sistem menghapus tampilan jurnal sebelumnya dan kembali ke status awal (Halaman Beranda)	Berhasil

Tahap selanjutnya adalah mengevaluasi tingkat akurasi sistem dalam menentukan skor dan label kemiripan semantik antar jurnal. Evaluasi disajikan secara terperinci pada Tabel 4, yang mencakup informasi mengenai masing-masing pasangan jurnal, serta label kemiripan dan skor yang dihasilkan oleh sistem.

Tabel 4. Evaluasi Akurasi Skor dan Label Kemiripan

Test Case	Skenario Pengujian	Skor	Label
Uji Jurnal Tidak Relevan	Menguji dua jurnal dari topik yang sama sekali berbeda (kanker dan ekosistem air)	10.00%	Tidak Relevan
Uji Jurnal Sedikit Berkaitan	Menguji pasangan jurnal yang memiliki hubungan lemah atau konteks berbeda dalam domain aplikasi atau teknik, namun masih dalam lingkup umum teknologi.	19.00%	Sedikit Berkaitan

Uji Jurnal Cukup Berkaitan	Sistem diuji dengan dua jurnal bertopik teknologi informasi yang menerapkan *sentence embedding* dan teknik klasifikasi berbasis vektor.	30.00%	Cukup Berkaitan
Uji Jurnal Sangat Berkaitan	Menguji kesamaan semantik antara dua jurnal dengan topik dan pendekatan yang sangat serupa	75.00%	Sangat Berkaitan

Hasil pengujian menunjukkan bahwa sistem mampu melakukan klasifikasi terhadap berbagai tingkat kemiripan abstrak jurnal. Skor kemiripan yang ditampilkan juga memiliki koherensi terhadap konteks semantik dari masing-masing pasangan jurnal. Hal ini membuktikan bahwa integrasi model *Sentence-Transformers* dengan algoritma *K-Means* dan penetapan label berbasis *centroid* mampu memberikan hasil yang representatif dalam konteks analisis komparatif jurnal ilmiah.

D. KESIMPULAN DAN SARAN

Penelitian ini berhasil mengembangkan sebuah sistem klasifikasi keterkaitan antarjurnal ilmiah dengan pendekatan *unsupervised learning* menggunakan model *Sentence-Transformers* dan algoritma *K-Means*. Sistem ini memungkinkan pengguna untuk membandingkan dua artikel jurnal ilmiah dalam format PDF secara otomatis, menghasilkan skor kemiripan semantik, klasifikasi tingkat keterkaitan ke dalam empat kategori, serta narasi AI sebagai penjelasan. Selain menghasilkan klasifikasi yang akurat, sistem ini juga membantu menghemat waktu dalam menilai keterkaitan antar jurnal yang sebelumnya dilakukan secara manual. Berdasarkan evaluasi terhadap beberapa metrik seperti *inertia, silhouette score*, dan *Davies-Bouldin Index*, nilai $k = 4$ ditentukan sebagai jumlah klaster optimal. Implementasi sistem berbasis web berjalan dengan baik dan hasil pengujian fungsional menunjukkan bahwa seluruh fitur utama bekerja sesuai harapan.

Saran untuk penelitian selanjutnya adalah melakukan perluasan pada jumlah dan keragaman data jurnal agar sistem dapat menggeneralisasi lebih baik. Selain itu, pengembangan klasifikasi berbasis *supervised learning* dan integrasi fitur perbandingan visual atau interaktif dapat menjadi alternatif peningkatan sistem di masa mendatang.

E. REFERENSI

[1] L. Bornmann, R. Haunschild, and R. Mutz, “Growth rates of modern science: a latent piecewise growth curve approach to model publication numbers from established and new literature databases,” *Humanit. Soc. Sci. Commun.*, vol. 8, no. 1, 2021, doi: 10.1057/s41599-021-00903-w.

[2] M. T. Colangelo, M. Meleti, S. Guizzardi, E. Calciolari, and C. Galli, “A Comparative Analysis of Sentence Transformer Models for Automated Journal Recommendation Using PubMed Metadata,” *Big Data Cogn. Comput.*, vol. 9, no. 3, pp. 1–18, 2025, doi: 10.3390/bdcc9030067.

[3] R. Kusumaningrum, S. F. Khoerunnisa, K. Khadijah, and M. Syafrudin, “Exploring Community

Awareness of Mangrove Ecosystem Preservation through Sentence-BERT and K-Means Clustering," *Inf.*, vol. 15, no. 3, pp. 1–14, 2024, doi: 10.3390/info15030165.

[4] H. T. A. Simanjuntak, P. E. P. Silaban, J. K. S. Manurung, and V. H. Sormin, "Klasterisasi Berita Bahasa Indonesia Dengan Menggunakan K-Means Dan Word Embedding," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 3, pp. 641–652, 2023, doi: 10.25126/jtiik.20231026468.

[5] A. Aszani, H. I. Wicaksono, U. Nadzima, and L. Heryawan, "Information Retrieval for Early Detection of Disease Using Semantic Similarity," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 17, no. 1, p. 45, 2023, doi: 10.22146/ijccs.80077.

[6] A. Subakti, H. Murfi, and N. Hariadi, "The performance of BERT as data representation of text clustering," *J. Big Data*, vol. 9, no. 1, 2022, doi: 10.1186/s40537-022-00564-9.

[7] Y. Ortakci, "Engineering Science and Technology , an International Journal Revolutionary text clustering : Investigating transfer learning capacity of SBERT models through pooling techniques," *Eng. Sci. Technol. an Int. J.*, vol. 55, no. April, p. 101730, 2024, doi: 10.1016/j.jestch.2024.101730.

[8] M. H. Weng, S. Wu, and M. Dyer, "Identification and Visualization of Key Topics in Scientific Publications with Transformer-Based Language Models and Document Clustering Methods," *Appl. Sci.*, vol. 12, no. 21, 2022, doi: 10.3390/app122111220.

[9] C. Y. Sy, L. L. Maceda, and M. B. Abisado, "AI-driven analysis: optimizing tertiary education policy through machine learning insights," *Int. J. Adv. Intell. Informatics*, vol. 10, no. 2, pp. 296–316, 2024, doi: 10.26555/ijain.v10i2.1525.

[10] R. Anggrainingsih, E. S. Wihidayat, and B. Widoyono, "Sentence embedding to improve rumour detection performance model," *IAES Int. J. Artif. Intell.*, vol. 13, no. 1, pp. 115–121, 2024, doi: 10.11591/ijai.v13.i1.pp115-121.

[11] Octavian Ery Pamungkas *et al.*, "Classification of Rupiah to Help Blind with The Convolutional Neural Network Method," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 2, pp. 259–268, 2022, doi: 10.29207/resti.v6i2.3852.

[12] Irbah salsabila and Yuliant Sibaroni, "Multi Aspect Sentiment of Beauty Product Reviews using SVM and Semantic Similarity," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 3, pp. 520–526, 2021, doi: 10.29207/resti.v5i3.3078.

[13] S. Afriyani, S. Surono, and M. I. Solihin, "Chi-Square Feature Selection with Pseudo-Labelling in Natural Language Processing," vol. 8, no. 3, pp. 896–909, 2024.

[14] Z. Zainuddin and A. A. N. Risal, "Balanced clustering for student admission school zoning by parameter tuning of constrained k-means," *IAES Int. J. Artif. Intell.*, vol. 13, no. 2, pp. 2299–2311, 2024,

doi: 10.11591/ijai.v13.i2.pp2301-2313.

[15] E. M. Hambi and F. Benabbou, “A deep learning based technique for plagiarism detection: A comparative study,” *IAES Int. J. Artif. Intell.*, vol. 9, no. 1, pp. 81–90, 2020, doi: 10.11591/ijai.v9.i1.pp81-90.

[16] R. Annisa, D. Rosiyadi, and D. Riana, “Improved point center algorithm for k-means clustering to increase software defect prediction,” *Int. J. Adv. Intell. Informatics*, vol. 6, no. 3, pp. 328–339, 2020, doi: 10.26555/ijain.v6i3.484.