Building of Informatics, Technology and Science (BITS)
Volume 7, No 3, December 2025 Page: 1963−1973
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i3.8859

Implemetasi TF-IDF N-Gram dan Algoritma Nearest Centroid untuk Klasifikasi Topik Tugas Akhir

Rohima Choirul Hana[1], Defri Kurniawan[2,*]

[1] Fakultas Ilmu Komputer, Program Studi Teknik Informatika, Universitas Dian Nuswantoro, Semarang, Indonesia
[2] Fakultas Ilmu Komputer, Program Studi Magister Teknik Informatika, Universitas Dian Nuswantoro, Semarang, Indonesia
Email: [1]111202214603@mhs.dinus.ac.id, [2,*]defri.kurniawan@dsn.dinus.ac.id
Email Penulis Korespondensi: defri.kurniawan@dsn.dinus.ac.id
Submitted: **05/12/2025**; Accepted: **26/12/2025**; Published**: 26/12/2025**

Abstrak−Penelitian ini menghadirkan alur kurasi judul Tugas Akhir pada Program Studi Teknik Informatika yang ringan dan terjelaskan dengan memadukan TF-IDF n-gram (1–2) dan pengklasifikasi Nearest Centroid berbasis kosinus. Judul dikelompokkan ke tiga kelas bidang kajian internal, yaitu RPLD, SC, dan SKKKD, untuk membantu pengelompokan topik dan penentuan pembimbing. Pendekatan diwujudkan sebagai aplikasi web Streamlit yang mendukung unggah Excel dengan preview dan simpan persisten, standarisasi kolom, normalisasi teks, penolakan duplikasi berbasis judul yang dinormalisasi, pelatihan cepat pada data berlabel, prediksi topik untuk judul baru, serta temu kembali judul paling mirip untuk membantu kurasi. Kontribusi operasional utama adalah penautan langsung hasil klasifikasi ke daftar dosen per bidang yang dikelola program studi, sehingga mahasiswa dapat mengidentifikasi pembimbing yang sesuai dan koordinator memperoleh alur kerja yang lebih konsisten dan dapat diaudit. Pada korpus lintas semester berisi 1.057 judul, evaluasi stratified 5-fold cross-validation menunjukkan akurasi rata-rata 92,43 persen, Macro F1 0,875, Micro F1 0,924, dan Weighted F1 0,925 yang menegaskan keseimbangan antara ketepatan, efisiensi, dan keterjelasan pada skenario short-text. Penelusuran keputusan didukung oleh top terms per kelas dan daftar judul tetangga terdekat sehingga mudah diaudit oleh pengguna non teknis. Keterbatasan muncul pada kelas minor sehingga pekerjaan lanjut meliputi perluasan korpus berlabel, penambahan n-gram karakter, dan eksplorasi representasi hibrida ringan.

Kata Kunci: TF-IDF; Nearest Centroid; Klasifikasi Judul TA; Cosine Similarity; Streamlit

Abstract−This study presents a lightweight and explainable workflow for curating undergraduate thesis titles in the Informatics Engineering Study Program by combining TF-IDF n-gram (1–2) features with a cosine based Nearest Centroid classifier. Titles are grouped into three internal research area classes, RPLD, SC, and SKKKD, to support topic grouping and supervisor assignment. The approach is implemented as a Streamlit web application that supports Excel upload with preview and persistent saving, column standardization, text normalization, duplicate rejection using normalized titles, rapid training on labeled data, topic prediction for new titles, and retrieval of the most similar titles to assist curation. A key operational contribution is the direct linkage from predicted classes to the program maintained lecturer list for each area, enabling students to identify suitable supervisors and helping coordinators run a consistent and auditable workflow. On a multi semester corpus of 1,057 titles, stratified 5-fold cross-validation achieved 92.43 percent average accuracy, Macro F1 of 0.875, Micro F1 of 0.924, and Weighted F1 of 0.925, indicating a balance between accuracy, efficiency, and interpretability for short text. Decision inspection is supported by class specific top terms and nearest neighbor title lists. Limitations mainly stem from the minority class, therefore future work will expand labeled corpora, add character level n grams, and explore lightweight hybrid representations.

Keywords: TF-IDF; Nearest Centroid; Thesis Title Classification; Cosine Similarity; Streamlit

1. PENDAHULUAN

Arsip judul Tugas Akhir di program studi terus bertambah pada setiap semester sehingga proses pengelompokan topik dan pemeriksaan kemiripan judul yang dilakukan secara manual menjadi semakin berat. Penilai harus membaca satu per satu judul, membandingkannya dengan arsip sebelumnya, dan memastikan tidak terjadi duplikasi topik maupun ketidakseimbangan distribusi bidang kajian. Pada konteks *short text* seperti judul Tugas Akhir, informasi leksikal yang tersedia sangat terbatas sehingga representasi menjadi *sparse* dan mudah kehilangan nuansa semantik [1], [2]. Kondisi ini membuat proses kurasi manual rawan subjektivitas, sulit diaudit ketika volume dokumen meningkat, dan tidak mudah direplikasi oleh koordinator baru. Di sisi lain, program studi membutuhkan alur kerja yang cepat, konsisten, dan transparan agar layanan akademik kepada mahasiswa tetap terjaga kualitasnya [3]–[5].

Penggunaan n-gram pada judul yang bersifat *short text* diperlukan karena banyak makna topik justru muncul sebagai frasa dua kata (kolokasi), misalnya "sistem informasi", "*data mining*", atau "*deep learning*". Jika hanya unigram yang digunakan, frasa tersebut terpecah menjadi token-token umum yang mudah muncul lintas bidang kajian sehingga meningkatkan ambiguitas. Dengan memasukkan unigram–bigram, representasi TF-IDF dapat menangkap konteks lokal dan pola frasa yang lebih spesifik, sehingga batas antar kelas menjadi lebih tegas tanpa menambah kompleksitas komputasi secara signifikan [1], [2].

Beragam penelitian sebelumnya menunjukkan bahwa representasi berbasis *Term Frequency–Inverse Document Frequency (TF-IDF)* dan ukuran kemiripan vektor seperti *cosine similarity* merupakan fondasi yang kuat untuk pengolahan *short text* karena sederhana, efisien, dan cukup stabil terhadap perubahan korpus [1], [2], [6], [7]. Pada ranah Tugas Akhir, beberapa studi memanfaatkan kombinasi TF-IDF dan Support Vector Machine (SVM) atau Naïve Bayes untuk mengklasifikasikan topik dan membantu kurasi arsip judul [3], [8]. Penelitian lain menerapkan TF-IDF dan *cosine similarity* untuk mengukur kemiripan judul sehingga potensi duplikasi judul dapat diidentifikasi lebih cepat [4], [5]. Pendekatan berbasis *ensemble classifier* dan pemilihan fitur seperti *Chi-Square* termodifikasi juga

This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 3, December 2025 Page: 1963−1973
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i3.8859

dilaporkan mampu meningkatkan akurasi klasifikasi judul Tugas Akhir dengan biaya komputasi yang masih dapat diterima di lingkungan kampus [9].

Di luar ranah Tugas Akhir, TF-IDF secara luas digunakan sebagai *baseline* untuk berbagai tugas klasifikasi teks, baik pada bahasa Indonesia maupun bahasa lain. Kajian mengenai prapemrosesan *short text* pada media sosial dan teks mikro menegaskan bahwa normalisasi, penghapusan noise, dan penanganan istilah khusus mempunyai dampak signifikan terhadap kualitas fitur dan kinerja klasifikasi [7], [10]. Berbagai penelitian juga menunjukkan bahwa penggabungan TF-IDF dengan arsitektur *deep learning* seperti Bidirectional *Long Short-Term Memory (Bi-LSTM)* dan *Convolutional Neural Network (CNN–LSTM)* dapat meningkatkan kemampuan model dalam menangkap konteks lokal sekaligus dependensi panjang, baik untuk teks pendek maupun panjang [10], [11], [12], [13], [14]. Namun, pendekatan ini umumnya membutuhkan sumber daya komputasi yang lebih besar dan pipeline yang lebih kompleks.

Pada sisi *state of the art*, model pralatih berbasis *transformer* seperti IndoBERT telah menunjukkan kinerja unggul pada berbagai tugas *Natural Language Processing (NLP)* bahasa Indonesia, misalnya klasifikasi sentimen, analisis multi-label, dan ekstraksi informasi [10], [11], [12], [15]. Studi komparatif memperlihatkan bahwa IndoBERT umumnya melampaui model sekuensial klasik seperti LSTM dari sisi akurasi, tetapi memerlukan proses *fine-tuning*, pengelolaan parameter, dan sumber daya komputasi yang tidak selalu sejalan dengan kondisi infrastruktur di banyak program studi [11], [16]–[18]. Untuk unit pengelola akademik dengan sumber daya terbatas, tuntutan komputasi dan kompleksitas perawatan sistem menjadikan adopsi penuh *transformer* kurang ideal, terutama bila alat perlu dijalankan dan diaudit oleh staf non-teknis dalam jangka panjang [3], [13].

Berbagai buku ajar juga menempatkan TF-IDF dan representasi berbasis *bag-of-words* sebagai landasan penting dalam penambangan teks dan klasifikasi dokumen. Aggarwal membahas TF-IDF dan variasi representasi teks secara mendalam dalam buku *Machine Learning for Text* [19], sementara Raschka dan Mirjalili menjelaskan penggunaan TF-IDF dan pengklasifikasi berbasis vektor dengan pustaka scikit-learn dalam buku *Machine Learning with PyTorch and Scikit-Learn* [20]. Buku *Blueprints for Text Analytics Using Python* juga menampilkan berbagai pola desain sistem analitik teks berbasis TF-IDF dan teknik pembelajaran mesin yang serupa dengan yang digunakan pada penelitian ini [21]. Rujukan-rujukan tersebut menegaskan bahwa meskipun pendekatan *deep learning* dan *transformer* semakin dominan, kombinasi TF-IDF dan model klasifikasi yang ringan tetap relevan ketika *reproducibility*, transparansi, dan efisiensi komputasi menjadi prioritas.

Dalam ranah pengklasifikasi teks berbasis representasi vektor, berbagai penelitian mengembangkan variasi pembobotan fitur, penanganan ketidakseimbangan kelas, dan skema tetangga terdekat untuk meningkatkan stabilitas dan akurasi pada data berdimensi tinggi [16], [22], [23]. Pendekatan ini menarik karena hanya menyimpan *centroid* tiap kelas dan keputusan dapat dijelaskan melalui kontribusi fitur pada *centroid*. Namun, penerapan pendekatan *centroid* yang ringan dan *explainable* pada konteks judul Tugas Akhir berbahasa Indonesia masih relatif terbatas dibandingkan pendekatan yang didominasi SVM, Naïve Bayes, atau ensemble pohon keputusan [3], [8]–[10]. Kajian tersebut muncul beberapa kesenjangan. Penelitian judul Tugas Akhir sebelumnya umumnya berfokus pada klasifikasi topik atau pemeriksaan kemiripan saja, belum mengaitkan keduanya dalam satu alur kerja kurasi yang utuh [3]–[5], [9]. Integrasi antara hasil klasifikasi topik dengan proses rekomendasi dosen pembimbing juga belum banyak dibahas secara eksplisit, padahal hal ini sangat relevan untuk operasional program studi. Selain itu, solusi berbasis *transformer* dan *deep learning* menawarkan akurasi tinggi tetapi kurang sejalan dengan kebutuhan sistem yang ringan dan mudah direplikasi di lingkungan dengan keterbatasan sumber daya komputasi [16]–[20].

Penelitian ini berupaya menjembatani kebutuhan tersebut dengan merancang alur kurasi judul Tugas Akhir yang memadukan TF-IDF n-gram dan pengklasifikasi *Nearest Centroid* berbasis *cosine similarity* dalam sebuah aplikasi *web*. Sistem mendukung unggah berkas Excel dengan pratinjau dan simpan persisten, standarisasi kolom, normalisasi teks, deduplikasi judul, pelatihan cepat pada data berlabel, prediksi bidang kajian, temu kembali judul paling mirip, serta penautan label hasil klasifikasi ke daftar dosen per bidang kajian. Dengan demikian, klasifikasi topik, deteksi kemiripan judul, dan rekomendasi dosen pembimbing diikat dalam satu antarmuka operasional yang ringan, *explainable*, dan mudah diaudit [3]–[5], [9], [21]–[23], [26].

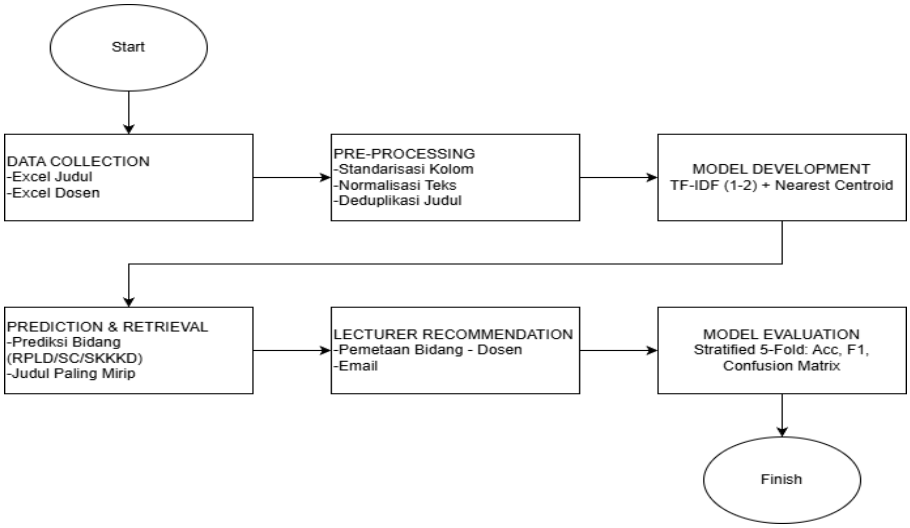
2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Tahapan penelitian pada studi ini menggambarkan urutan penerapan metode dari awal hingga evaluasi, seperti ditunjukkan pada Gambar 1. Selain menjelaskan alur teknis, tahapan ini juga dirancang agar mudah direplikasi pada lingkungan laboratorium standar serta menghasilkan keluaran yang konsisten untuk mendukung proses kurasi di unit akademik. Penelitian dimulai dari pengumpulan data judul Tugas Akhir dan data dosen dalam bentuk file *Excel*, kemudian dilanjutkan dengan prapemrosesan berupa standarisasi kolom, normalisasi teks, dan deduplikasi judul. Setelah itu dilakukan pengembangan model dengan membentuk fitur TF-IDF n-gram dan melatih algoritma *Nearest Centroid*. Model yang dihasilkan digunakan untuk tahap prediksi bidang kajian serta temu kembali judul paling mirip, yang selanjutnya menjadi dasar pemberian rekomendasi dosen pembimbing. Tahap terakhir adalah evaluasi model menggunakan skema stratified *5-fold cross-validation* dan metrik akurasi, F1, serta *confusion matrix* untuk memastikan metode yang digunakan menghasilkan kinerja sesuai harapan.

Copyright © 2025 **Author**, Page 1964
This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 3, December 2025 Page: 1963−1973
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i3.8859



Gambar 1. Tahapan Penelitian

2.2 Data Collection

Pengumpulan data dilakukan dengan memanfaatkan arsip judul Tugas Akhir dari beberapa semester akademik pada program studi Teknik Informatika. Arsip disediakan dalam bentuk berkas *Excel* yang minimal memuat kolom judul dan label bidang kajian. Melalui aplikasi, pengguna dapat mengunggah berkas tersebut melalui antarmuka *web*, kemudian sistem akan menampilkan pratinjau isi berkas sebelum disimpan secara permanen. Langkah ini memudahkan operator untuk memastikan struktur kolom dan isi data sudah sesuai kebutuhan.

Seluruh berkas dari empat semester, yaitu Genap 2022/2023, Ganjil 2022/2023, Ganjil 2023/2024, dan Genap 2023/2024, digabungkan menjadi satu *dataset*. Setelah proses standarisasi dan pembersihan awal, diperoleh total 1.057 judul Tugas Akhir. Ringkasan jumlah judul per semester disajikan pada Tabel 1. Penggunaan arsip internal sebagai sumber data sejalan dengan penelitian-penelitian sebelumnya yang memanfaatkan judul Tugas Akhir untuk klasifikasi topik maupun pendeteksian kemiripan judul, sehingga model yang dibangun benar-benar merefleksikan kebutuhan operasional program studi [3]–[5], [8], [9].

Pada penelitian ini, label bidang kajian diseragamkan menjadi tiga kelas, yaitu RPLD, SC, dan SKKKD. Untuk memberi konteks bagi pembaca di luar program studi, RPLD umumnya memuat judul dengan fokus rekayasa perangkat lunak dan pengembangan sistem, SC memuat topik komputasi cerdas seperti *machine learning*, *computer vision*, atau *natural language processing*, sedangkan SKKKD memuat topik sistem komputer, jaringan, dan keamanan. Distribusi data setelah pembersihan adalah 498 judul (RPLD), 542 judul (SC), dan 17 judul (SKKKD).

Tabel 1. Ringkasan jumlah judul Tugas Akhir per semester

Semester/Dataset	Jumlah Judul
Genap 2022/2023	336
Ganjil 2022/2023	204
Ganjil 2023/2024	248
Genap 2023/2024	269
Total	1.057

2.3 Preprocessing

Pada penelitian ini, *pre-processing* difokuskan pada tiga kegiatan utama, yaitu standarisasi kolom, normalisasi teks, dan deduplikasi judul. Ketiga langkah ini bertujuan mengubah data mentah dari berkas *Excel* menjadi korpus judul yang bersih, seragam, dan siap direpresentasikan ke dalam ruang vektor TF-IDF. Pendekatan tersebut sejalan dengan praktik umum prapemrosesan pada penelitian klasifikasi teks dan *short text* [1], [2], [7], [11], [21]–[23].

Langkah pertama adalah standarisasi kolom. Berkas *Excel* yang diunggah pengguna dapat memiliki nama kolom yang berbeda-beda, misalnya “judul”, “judul ta”, atau “*title*” untuk kolom judul, serta variasi lain untuk kolom bidang kajian. Agar proses selanjutnya tidak bergantung pada nama kolom asli, sistem memetakan seluruh variasi tersebut ke nama baku, seperti “Judul TA” untuk judul dan “Bid_Kajian” untuk label. Dengan cara ini, *pipeline* pengolahan data menjadi lebih sederhana dan konsisten meskipun struktur berkas sumber tidak seragam [3]–[5].

Setelah struktur kolom seragam, dilakukan normalisasi teks. Seluruh judul diubah ke huruf kecil (*lowercase*), spasi ganda disederhanakan, dan karakter yang tidak relevan seperti simbol tertentu serta tanda baca berlebih dihapus. Normalisasi ini bertujuan mengurangi variasi penulisan yang tidak memengaruhi makna, sehingga *n-gram* yang dihasilkan pada tahap pembentukan fitur tidak tersebar ke banyak bentuk yang sebenarnya sama [1], [2], [7], [10]. Praktik normalisasi seperti ini banyak dianjurkan dalam literatur penambangan teks dan *short-text processing* [21]–[23].

Copyright © 2025 **Author**, Page 1965
This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 3, December 2025 Page: 1963−1973
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i3.8859

Langkah terakhir adalah deduplikasi judul. Setelah judul dinormalisasi, sistem membentuk representasi teks kanonik dan memeriksa apakah terdapat entri yang benar-benar sama. Jika ditemukan beberapa baris dengan judul identik, hanya satu entri yang dipertahankan, sementara sisanya dianggap duplikasi. Deduplikasi penting untuk mencegah bias pelatihan dan mengurangi risiko data leakage antara data latih dan uji, terutama ketika digunakan skema *cross-validation*. Selain itu, judul yang unik akan membuat proses analisis kemiripan judul dan evaluasi model menjadi lebih akurat [4], [5], [8], [14], [24].

Melalui rangkaian *pre-processing* ini, judul Tugas Akhir yang semula tersebar di berbagai berkas dengan penulisan beragam diubah menjadi korpus teks yang lebih bersih dan konsisten, sehingga siap dipetakan ke dalam representasi TF-IDF *n-gram* pada tahap pengembangan model berikutnya.

2.4 Model Development

Pada tahap *model development*, judul Tugas Akhir yang sudah melalui proses *pre-processing* direpresentasikan ke dalam vektor fitur menggunakan TF-IDF *n-gram* (1–2). Notasi yang digunakan adalah *t* untuk istilah, *d* untuk dokumen, *tf(t,d)* sebagai frekuensi kemunculan istilah *t* pada dokumen *d*, *df(t)* sebagai jumlah dokumen yang memuat *t*, dan *N* sebagai jumlah dokumen dalam korpus. Bobot term dihitung dengan skema *sublinear term frequency* dan *inverse document frequency* berbasis log, kemudian vektor dinormalisasi dengan norma L2 [1], [2], [6], [7], [19], [20]. Secara matematis, pembobotan yang digunakan dapat dirangkum sebagai berikut.
Sublinear term frequency

$$tf'(t,d) = 1 + log\big(tf(t,d)\big) \tag{1}$$

Inverse document frequency

$$idf(t) = log(\frac{N}{df(t)}) \tag{2}$$

Bobot TF-IDF sebelum normalisasi

$$w(t,d) = tf'(t,d)idf(t) \tag{3}$$

Normalisasi L2 vektor dokumen

$$\widehat{\mathrm{w}}_d = \frac{\mathrm{w}_d}{\|\mathrm{w}_d\|_2} \tag{4}$$

Di sini $\mathrm{w}_d$menyatakan vektor bobot untuk dokumen *d*, sedangkan $\widehat{\mathrm{w}}_d$adalah vektor yang sudah ternormalisasi. Representasi TF-IDF ternormalisasi ini kemudian digunakan sebagai masukan untuk algoritma *Nearest Centroid*. Untuk setiap kelas *c* (RPLD, SC, SKKKD), dihitung *centroid* $\mu_c$sebagai rata-rata vektor dokumen latih pada kelas tersebut:

$$\mu_c = \frac{1}{|S_c|} \sum_{d \in S_c} \widehat{\mathrm{w}}_d \tag{5}$$

dengan $S_c$adalah himpunan dokumen latih pada kelas *c*. Prediksi label untuk judul baru dengan vektor $\widehat{\mathbf{w}}_{d^*}$ dilakukan dengan memilih kelas yang menghasilkan nilai *dot product* terbesar terhadap *centroid* kelas. Secara matematis, label prediksi $\hat{y}$ diperoleh dengan:

$$\hat{y} = arg_{c \in \{RPLD, SC, SKKKD\}} \max (\widehat{\mathrm{w}}_{d^*} \cdot \mu_c) \tag{6}$$

Karena seluruh vektor TF-IDF telah dinormalisasi dengan norma L2, nilai *dot product* pada persamaan tersebut ekuivalen dengan nilai *cosine similarity* antara judul yang akan diprediksi dan *centroid* masing-masing kelas. Pendekatan berbasis *centroid* ini dipilih karena kebutuhan komputasinya ringan, mudah diimplementasikan menggunakan *scikit-learn*, dan keputusan model dapat dijelaskan melalui istilah-istilah dengan bobot tertinggi pada *centroid* masing-masing kelas sehingga selaras dengan tuntutan sistem yang transparan dan mudah diaudit [9], [19], [20], [22].

2.5 Prediction & Retrieval

Pada tahap *prediction & retrieval*, model yang telah dikembangkan mulai digunakan untuk memproses judul baru yang dimasukkan melalui antarmuka aplikasi. Setiap judul terlebih dahulu melewati rangkaian *pre-processing* yang sama seperti pada data latih, kemudian direpresentasikan ke dalam vektor TF-IDF *n-gram* (1–2). Dengan cara ini, bentuk fitur antara data latih dan data uji tetap konsisten sehingga hasil prediksi lebih dapat dipercaya [1], [2], [6], [7], [19], [20].

Setelah vektor TF-IDF judul baru diperoleh, sistem menjalankan proses prediksi bidang. Nilai kemiripan kosinus antara vektor judul dan *centroid* untuk masing-masing kelas (RPLD, SC, SKKKD) dihitung, lalu kelas dengan nilai kemiripan tertinggi dipilih sebagai label prediksi. Nilai kemiripan ini sekaligus dapat ditampilkan sebagai ukuran keyakinan model terhadap prediksinya. Mekanisme ini mengikuti prinsip dasar *Nearest Centroid* yang banyak digunakan pada tugas klasifikasi teks karena sederhana, efisien, dan mudah dijelaskan [9], [19], [20], [22].

This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 3, December 2025 Page: 1963−1973
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i3.8859

Selain memberikan label bidang, sistem juga melakukan pencarian judul paling mirip (*retrieval*). Vektor TF-IDF judul baru dibandingkan dengan seluruh vektor judul yang ada di korpus menggunakan kemiripan kosinus. Beberapa judul dengan nilai kemiripan tertinggi kemudian ditampilkan sebagai daftar judul serupa. Informasi ini membantu operator mengecek kemungkinan adanya kemiripan atau duplikasi judul, sekaligus menilai kewajaran hasil klasifikasi yang diberikan model [4], [5], [8], [24].

Dengan demikian, tahap *prediction & retrieval* menjadi jembatan antara model yang dibangun secara algoritmik dan kebutuhan operasional di program studi, yaitu penentuan bidang kajian judul baru dan identifikasi cepat terhadap judul-judul yang memiliki kedekatan tema di dalam arsip.

2.6 Lecturer Recommendation

Tahap *lecturer recommendation* memanfaatkan label bidang kajian hasil prediksi untuk membantu penentuan dosen pembimbing yang sesuai. Data dosen disimpan dalam berkas Excel terpisah yang berisi informasi nama, bidang keahlian utama, serta alamat *email*. Setelah sistem menetapkan label bidang untuk suatu judul Tugas Akhir (RPLD, SC, atau SKKKD), label tersebut digunakan sebagai kunci untuk melakukan pemetaan ke daftar dosen yang memiliki bidang keahlian yang sama. Dengan cara ini, proses pencarian calon pembimbing tidak lagi dilakukan secara manual, tetapi dibantu oleh sistem berdasarkan hasil klasifikasi yang telah diperoleh [3], [8], [25].

Pada implementasi ini, setiap dosen direpresentasikan oleh atribut "bidang keahlian utama" pada berkas *Excel* dosen sebagai label tunggal untuk pemetaan. Jika pada suatu bidang terdapat lebih dari satu dosen, sistem menampilkan seluruh dosen yang cocok, untuk menjaga keluaran deterministik, daftar ditampilkan dalam urutan alfabetis. Hasil pemetaan bidang–dosen ditampilkan kepada pengguna dalam bentuk daftar rekomendasi yang memuat nama dosen dan alamat *email*-nya, sehingga koordinator maupun mahasiswa dapat segera melihat siapa saja dosen yang relevan dengan topik yang diajukan. Sistem tidak menggantikan keputusan akhir manusia, tetapi berfungsi sebagai *decision support* yang mempercepat proses dan menjaga konsistensi antara bidang topik dengan keahlian dosen. Pendekatan ini sejalan dengan konsep sistem pendukung keputusan berbasis klasifikasi teks yang memanfaatkan keluaran model sebagai dasar rekomendasi, sebagaimana banyak dibahas dalam literatur penambangan teks dan *machine learning* terapan [21]–[23].

Selain menampilkan rekomendasi, hasil pemetaan juga dapat disimpan sebagai riwayat agar pada tahap evaluasi berikutnya program studi dapat menelusuri kembali hubungan antara judul, bidang kajian, dan dosen pembimbing yang dipilih. Informasi ini berguna untuk melihat pola beban pembimbingan, kecocokan topik dengan keahlian dosen, serta perbaikan sistem rekomendasi pada pengembangan penelitian selanjutnya.

2.7 Model Evaluation

Kinerja model dievaluasi menggunakan skema *stratified 5-fold cross-validation* pada seluruh data berlabel. Pada setiap lipatan, model TF-IDF dan *Nearest Centroid* dilatih pada data latih dan diuji pada data uji, kemudian hasil seluruh lipatan dirata-rata. Evaluasi dilakukan dengan menyusun *confusion matrix* dan menghitung akurasi, *precision*, *recall*, serta F1-score untuk setiap kelas [1], [11], [21]–[23], [26]. *Confusion matrix* berdimensi $K \times K$(dengan K jumlah kelas) dinyatakan sebagai:

$$CM_{ij} = \{d \mid y(d) = i,\ \hat{y}(d) = j\} \tag{7}$$

dengan $y(d)$adalah label sebenarnya dan $\hat{y}(d)$adalah label hasil prediksi untuk dokumen d. Akurasinya dihitung sebagai proporsi prediksi yang benar terhadap seluruh data:

$$Acc = \frac{1}{N} \sum_{i=1}^{K} CM_{ii} \tag{8}$$

Untuk setiap kelas ke-i, didefinisikan:

$$TP_i = CM_{ii},\ FP_i = \sum_{j \neq i} CM_{ji},\ FN_i = \sum_{j \neq i} CM_{ij} \tag{9}$$

Sehingga *precision*, *recall*, dan F1-score per kelas dirumuskan sebagai:

$$Prec_i = \frac{TP_i}{TP_i + FP_i},\ Rec_i = \frac{TP_i}{TP_i + FN_i},\ F1_i = \frac{2Prec_i\, Rec_i}{Prec_i + Rec_i} \tag{10}$$

Dari nilai F1 per kelas kemudian dapat dihitung varian *macro*, *micro*, maupun *weighted* F1 sesuai kebutuhan analisis. Rangkaian ukuran ini memberikan gambaran yang lebih lengkap tentang performa model, khususnya pada kondisi distribusi kelas yang tidak seimbang seperti pada kelas SKKKD, dan mengikuti praktik evaluasi yang lazim digunakan pada penelitian klasifikasi teks pendek [11], [21]–[23], [26].

3. HASIL DAN PEMBAHASAN

3.1 Ringkasan Kinerja Model

Evaluasi kinerja dilakukan menggunakan skema *stratified 5-fold cross-validation* pada 1.057 judul Tugas Akhir yang telah melalui proses prapemrosesan dan kanonisasi label. Pada setiap lipatan, model TF-IDF *n-gram* dengan algoritma

Copyright © 2025 **Author**, Page 1967
This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 3, December 2025 Page: 1963−1973
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i3.8859

Nearest Centroid dilatih pada data latih dan diuji pada data uji, kemudian hasil seluruh lipatan dirata-ratakan. Ringkasan nilai akurasi dan F1-score yang diperoleh ditunjukkan pada Tabel 2.

Tabel 2. Ringkasan kinerja model klasifikasi

Metrik Umum	Nilai
Akurasi rata-rata	92,43% ± 1,20%
Macro-F1	0,875
Micro-F1	0,924
Weighted-F1	0,925

Berdasarkan Tabel 2, model menghasilkan akurasi rata-rata sebesar 92,43% dengan deviasi sekitar 1,20%. Deviasi yang relatif kecil ini menunjukkan variasi performa antarlipatan rendah, sehingga model cenderung konsisten dan tidak sensitif terhadap pembagian data pada skema *cross-validation*. Nilai *Macro-F1* yang diperoleh sebesar 0,875, sedangkan *Micro-F1* dan *Weighted-F1* masing-masing bernilai 0,924 dan 0,925. Kombinasi metrik tersebut menunjukkan bahwa model tidak hanya memiliki ketepatan klasifikasi secara keseluruhan yang tinggi, tetapi juga cukup seimbang dalam menangani perbedaan jumlah data pada masing-masing kelas. Dengan demikian, pendekatan TF-IDF *n-gram* yang dipadukan dengan *Nearest Centroid* dapat dianggap layak digunakan sebagai alat bantu kurasi judul Tugas Akhir di lingkungan program studi.

3.2 Kinerja Model per Kelas

Kinerja model secara lebih rinci untuk masing-masing kelas ditunjukkan pada Tabel 3. Tabel tersebut memuat nilai *precision, recall,* F1-score, serta jumlah data (*support*) untuk tiga kelas kanonik, yaitu RPLD, SC, dan SKKKD.

Tabel 3. Kinerja model per kelas

Kelas	Precision	Recall	F1	Support
RPLD	0,914	0,946	0,929	498
SC	0,948	0,904	0,925	542
SKKKD	0,652	0,938	0,770	17

Berdasarkan Tabel 3, kelas RPLD memiliki *precision* sebesar 0,914 dan *recall* 0,946, dengan F1-score 0,929. Hal ini menunjukkan bahwa sebagian besar judul yang sebenarnya termasuk RPLD berhasil dikenali oleh model, sementara jumlah salah positif pada kelas ini relatif kecil.

Pada kelas SC, *precision* mencapai 0,948 dan *recall* 0,904 dengan F1-score 0,925. Nilai precision yang tinggi mengindikasikan bahwa judul yang diprediksi sebagai SC hampir selalu benar, sehingga model jarang memberikan prediksi SC pada judul yang seharusnya termasuk kelas lain.

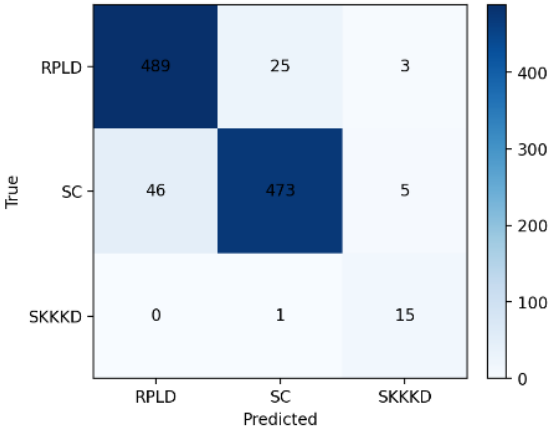
Kelas SKKKD memiliki *precision* 0,652 dan *recall* 0,938 dengan F1-score 0,770. Meskipun ukuran data kelas ini jauh lebih kecil dibanding dua kelas lainnya (hanya 17 judul), *recall* yang tinggi menunjukkan bahwa mayoritas judul SKKKD berhasil ditemukan oleh model. Nilai *precision* yang lebih rendah terutama dipengaruhi oleh adanya beberapa judul jaringan dan keamanan yang secara leksikal beririsan dengan topik *computer vision* atau *natural language processing*, sehingga sesekali muncul salah positif pada kelas SKKKD. Secara keseluruhan, pola ini mengonfirmasi bahwa model bekerja sangat baik pada dua kelas mayor (RPLD dan SC) dan tetap kompetitif untuk kelas minor SKKKD.

3.3 Confusion Matrix dan Pola Kekeliruan

Visualisasi distribusi prediksi model disajikan melalui Confusion Matrix pada Gambar 2, yang merangkum performa klasifikasi secara komprehensif. Secara keseluruhan, matriks ini menunjukkan dominasi nilai pada elemen diagonal utama, sebuah indikator krusial bahwa mayoritas judul dokumen telah berhasil dipetakan ke dalam label kelas yang tepat. Konsistensi ini selaras dengan perolehan nilai akurasi dan F1-Score yang tinggi sebagaimana dipaparkan pada Tabel 2 dan Tabel 3, sekaligus memberikan bukti empiris mengenai stabilitas performa model di seluruh kelas yang dievaluasi. Perlu dicatat bahwa Confusion Matrix pada Gambar 2 merupakan hasil akumulasi prediksi melalui skema Stratified 5-Fold Cross-Validation. Pendekatan ini memastikan bahwa setiap sel merepresentasikan frekuensi perpindahan dari label aktual ke label prediksi secara objektif berdasarkan distribusi data yang representatif. Dominasi angka pada garis diagonal menegaskan bahwa penggunaan fitur TF-IDF n-gram sangat efektif dalam mengekstraksi karakteristik leksikal yang mampu membedakan tiap bidang ilmu. Meskipun demikian, terdapat sejumlah kekeliruan klasifikasi yang terdeteksi pada sel non-diagonal. Fenomena ini mengindikasikan adanya tantangan inheren dalam pemrosesan bahasa alami (NLP), seperti penggunaan judul yang terlalu singkat (short text), adanya irisan istilah (oksimum) antar bidang yang tumpang tindih, serta pengaruh ketidakseimbangan kelas (class imbalance), khususnya pada kategori SKKKD yang memiliki jumlah sampel lebih sedikit. Oleh karena itu, interpretasi terhadap Confusion Matrix ini tidak dilakukan secara tunggal, melainkan diperkuat dengan analisis metrik per kelas pada Tabel 3. Hal ini bertujuan untuk mengidentifikasi arah kesalahan prediksi (misclassification bias) serta mengevaluasi sejauh mana dampak ketidakseimbangan data memengaruhi presisi pada kelas-kelas minor.

This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 3, December 2025 Page: 1963−1973
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i3.8859



Gambar 2. Confusion matrix hasil *stratified 5-fold cross-validation*

Di luar diagonal, kekeliruan paling sering terjadi pada pasangan kelas RPLD–SC. Judul yang menggabungkan kata kunci algoritmik, seperti *machine learning*, *deep learning*, atau *data mining*, dengan fokus implementasi sistem cenderung berada di wilayah abu-abu. Pada kasus tersebut, model sesekali mengelompokkan judul yang seharusnya RPLD ke SC ketika sinyal leksikal algoritma lebih kuat daripada konteks rekayasa perangkat lunak, atau sebaliknya ketika judul lebih menonjolkan sisi aplikasi dibanding sisi komputasional.

Pola kekeliruan lainnya muncul pada kelas SKKKD, khususnya ketika judul berkaitan dengan keamanan jaringan atau infrastruktur tetapi juga menyebut istilah yang umum di komputer cerdas, seperti *intrusion detection*, *anomaly detection*, atau istilah yang sering digunakan dalam *computer vision* dan *natural language processing*. Judul yang terlalu pendek atau sangat generik, misalnya hanya memuat kata kerja umum seperti “deteksi” atau “optimasi” tanpa konteks tambahan, menghasilkan vektor TF-IDF yang kurang informatif sehingga lebih mudah terseret ke *centroid* kelas mayor.

Secara keseluruhan, confusion matrix memperlihatkan bahwa kesalahan model terutama terjadi pada judul dengan konteks lintas domain atau judul yang sangat singkat. Informasi ini penting sebagai dasar perbaikan, baik pada sisi penulisan judul agar lebih kaya konteks maupun pada sisi model, misalnya dengan menambah fitur yang lebih peka terhadap istilah spesifik setiap bidang kajian.

3.4 Analisis Kesalahan

Analisis terhadap judul-judul yang salah klasifikasi menunjukkan bahwa sebagian besar kesalahan berkaitan dengan ambiguitas batas antarbidang kajian serta keterbatasan informasi pada judul. Kekeliruan yang paling sering muncul adalah perpindahan antara kelas RPLD dan SC. Banyak judul yang di satu sisi menonjolkan pengembangan aplikasi atau sistem informasi, tetapi di sisi lain juga memuat kata kunci algoritmik seperti *machine learning, deep learning*, atau *data mining*. Pada kasus seperti ini, model cenderung tertarik pada istilah algoritmik yang kuat sehingga vektor judul bergerak mendekati *centroid* kelas SC, meskipun secara substansi tugas akhirnya lebih dekat ke rekayasa perangkat lunak. Sebaliknya, judul yang fokus pada penerapan algoritma tertentu di dalam sistem sering kali dipetakan ke RPLD karena frasa mengenai perancangan dan implementasi sistem lebih dominan di dalam teks.

Kesalahan pada kelas SKKKD umumnya berkaitan dengan keterbatasan jumlah data dan irisan istilah dengan bidang lain. Beberapa judul yang berkaitan dengan jaringan komputer dan keamanan informasi tetap memiliki terminologi yang dekat dengan area komputer cerdas, misalnya pada topik deteksi serangan atau pemantauan trafik berbasis pembelajaran mesin. Ketika judul tidak memberikan konteks teknis yang cukup spesifik, model kesulitan membedakan apakah topik tersebut lebih tepat dikategorikan sebagai SKKKD atau SC. Kondisi ini diperkuat oleh fakta bahwa jumlah judul SKKKD jauh lebih sedikit dibandingkan dua kelas lainnya, sehingga variasi pola yang dapat dipelajari model untuk kelas ini menjadi terbatas.

Faktor lain yang berkontribusi terhadap kesalahan adalah sifat judul sebagai *short text* yang cenderung sangat ringkas. Judul yang hanya terdiri dari beberapa kata umum seperti “sistem deteksi serangan jaringan” atau “analisis kinerja jaringan” menghasilkan vektor TF-IDF yang relatif miskin fitur pembeda. Tanpa penjelasan tambahan mengenai pendekatan, platform, atau konteks implementasi, model lebih mengandalkan kata-kata generik yang dapat muncul di berbagai bidang kajian. Hal ini membuat batas antar kelas menjadi kurang tegas dan meningkatkan peluang salah klasifikasi, terutama pada judul yang berada di area perbatasan dua bidang.

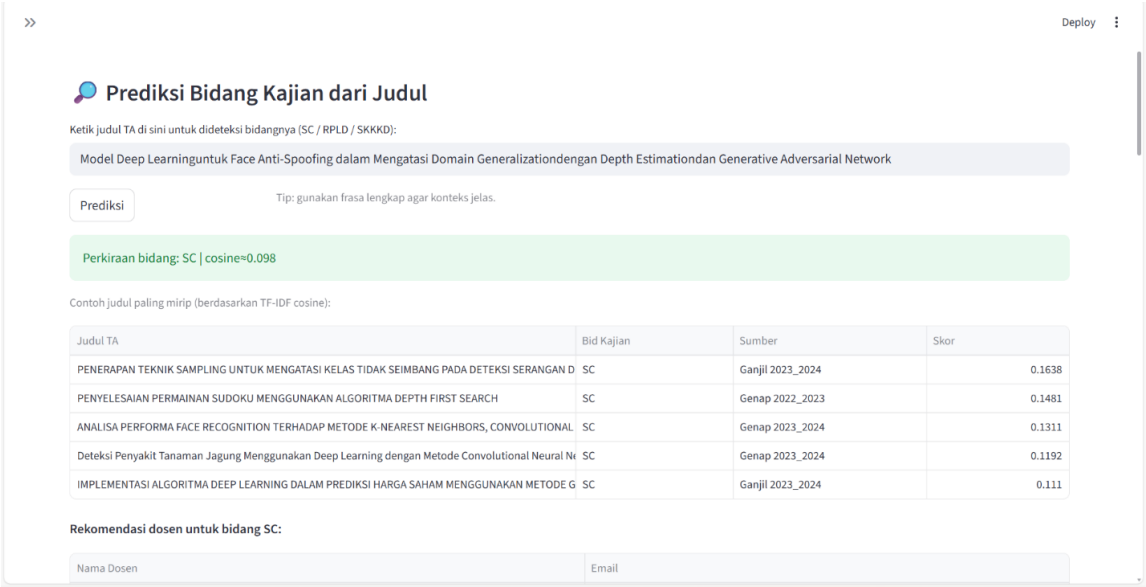
Temuan-temuan ini memberikan sejumlah masukan untuk pengembangan lebih lanjut. Dari sisi konten, penulisan judul dapat diarahkan agar memuat informasi yang sedikit lebih kaya, misalnya dengan menambahkan keterangan pendek mengenai jenis metode, domain aplikasi, atau lingkungan sistem yang digunakan. Dari sisi model, kualitas klasifikasi dapat ditingkatkan melalui penambahan data latih pada kelas yang masih minor, eksplorasi variasi fitur seperti *n-gram* karakter, atau penyesuaian aturan pasca-pemrosesan sederhana untuk judul yang mengandung istilah sangat generik. Dengan demikian, sistem diharapkan tidak hanya mempertahankan kinerja rata-rata yang tinggi, tetapi juga lebih andal pada kasus-kasus batas yang selama ini menjadi sumber kesalahan utama.

Copyright © 2025 **Author**, Page 1969
This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 3, December 2025 Page: 1963−1973
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i3.8859

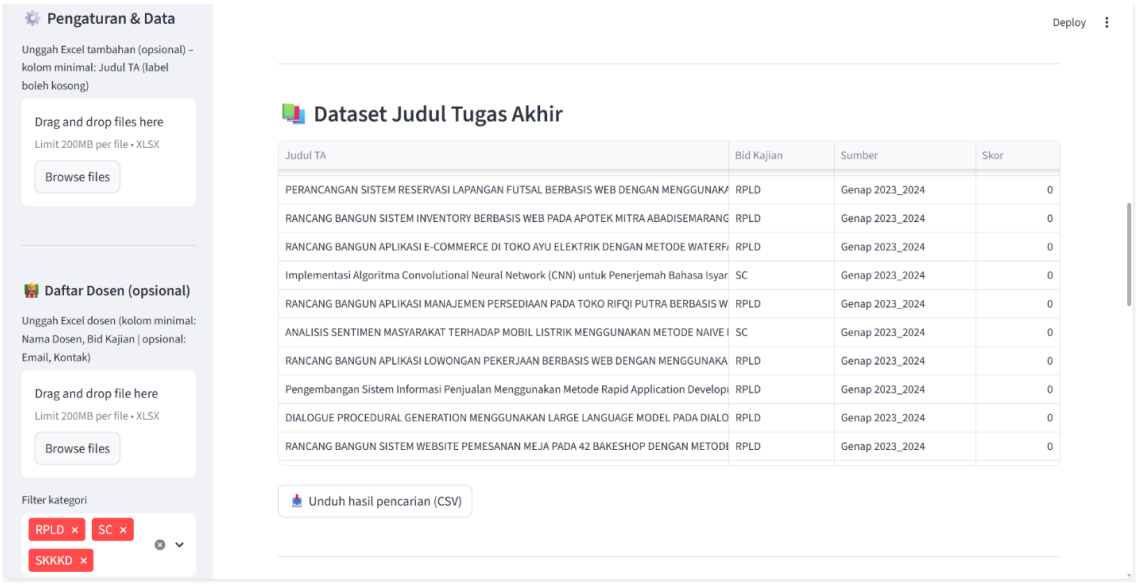
3.5 Implementasi Sistem

Sistem yang dikembangkan diwujudkan sebagai aplikasi *web* interaktif berbasis Python dan framework *Streamlit*. Antarmuka dirancang bersifat *task-oriented*, sehingga seluruh alur kerja—mulai dari unggah berkas, peninjauan data, pelatihan model, pengujian judul baru, hingga penelusuran judul mirip dan rekomendasi dosen—dapat dilakukan dalam satu tempat. Seluruh komponen yang tersedia pada aplikasi mengikuti tahapan dan parameter metodologi yang telah dijelaskan pada Bab 2, sehingga hasil evaluasi kuantitatif dapat direplikasi langsung melalui antarmuka tanpa perlu menjalankan skrip terpisah.



Gambar 3. Halaman prediksi dan rekomendasi dosen

Gambar 3 menampilkan halaman utama prediksi dan rekomendasi dosen. Pada halaman ini, pengguna memasukkan judul Tugas Akhir dalam sebuah isian teks, kemudian sistem menampilkan label bidang kajian yang diprediksi beserta skor kemiripan kosinus sebagai indikator tingkat keyakinan model. Berdasarkan label tersebut, aplikasi secara otomatis menautkan ke daftar dosen pada bidang yang sama dan menampilkan nama serta alamat surel yang relevan. Dengan cara ini, mahasiswa dapat segera mengidentifikasi calon pembimbing yang sesuai, sementara koordinator dapat melihat kembali hasil klasifikasi secara ringkas. Di bawah hasil prediksi, sistem juga menampilkan daftar judul paling mirip dari korpus yang telah tersimpan, sehingga keputusan model dapat diaudit melalui perbandingan dengan judul-judul yang sudah ada.



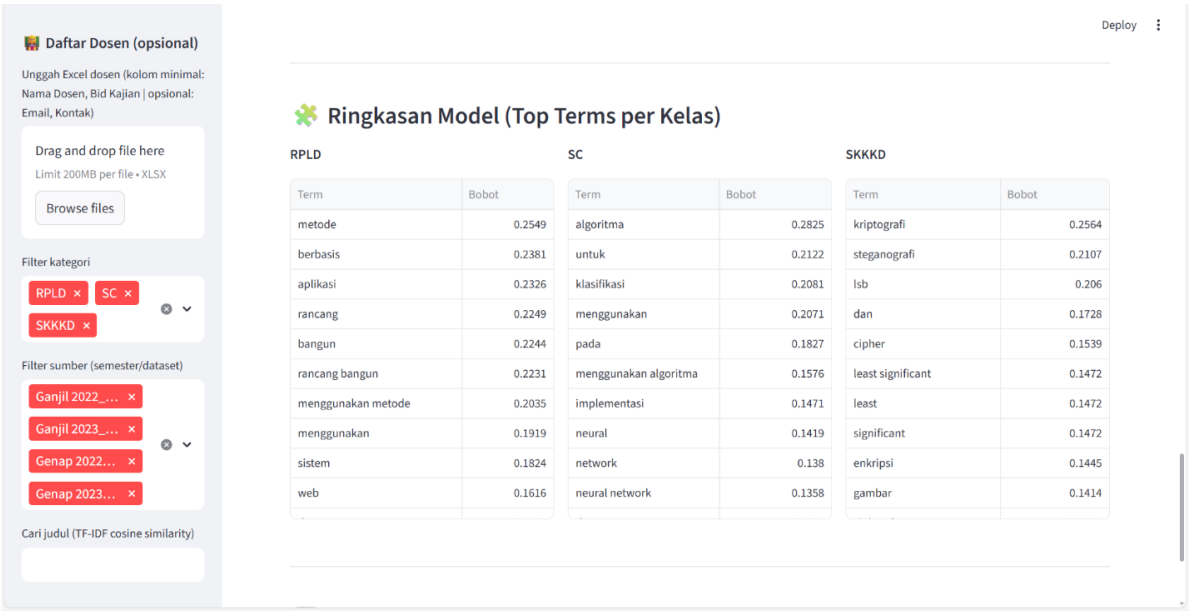
Gambar 4. Modul dataset dan unggah data

Gambar 4 memperlihatkan modul pengelolaan dataset dan unggah data. Modul ini menyediakan fasilitas untuk mengunggah berkas *Excel* yang berisi judul Tugas Akhir beserta label bidang kajian. Setelah berkas dipilih, aplikasi menampilkan pratinjau isi tabel agar pengguna dapat memastikan struktur kolom dan isi data sudah benar sebelum

Copyright © 2025 **Author**, Page 1970
This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 3, December 2025 Page: 1963−1973
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i3.8859

disimpan secara persisten. Pada tahap ini sistem melakukan standarisasi nama kolom, normalisasi teks judul, serta deduplikasi berbasis bentuk judul yang telah dinormalisasi, sehingga hanya satu entri unik yang dipertahankan untuk setiap judul. Ringkasan jumlah baris yang diterima dan ditolak ditampilkan di layar, sehingga operator dapat segera mengidentifikasi bila masih terdapat masalah pada data sumber.



Gambar 5. Ringkasan model berdasarkan *top terms* per kelas

Ringkasan model ditunjukkan pada Gambar 5. Halaman ini menampilkan daftar istilah dengan bobot tertinggi pada *centroid* masing-masing kelas dalam ruang TF-IDF. Informasi ini membantu pengguna memahami kata-kata apa saja yang paling mewakili setiap bidang kajian dari sudut pandang model. Tampilan tersebut melengkapi daftar tetangga terdekat pada halaman prediksi: pengguna dapat menelusuri alasan di balik suatu prediksi baik dari sisi fitur yang dominan (*top terms*) maupun dari sisi contoh judul yang paling mirip. Kombinasi kedua tampilan ini membuat proses penilaian dan audit keputusan model dapat dilakukan oleh pengguna nonteknis tanpa perlu melihat detail implementasi di tingkat kode.

Informasi *top terms* dapat digunakan untuk membantu memahami pola kesalahan pada *confusion matrix*. Secara umum, istilah yang paling menonjol pada kelas RPLD berkaitan dengan pengembangan sistem atau aplikasi, sedangkan kelas SC lebih banyak memuat istilah yang berhubungan dengan komputasi cerdas dan pemodelan. Pada beberapa judul yang memadukan istilah "sistem" dengan istilah algoritmik (misalnya "klasifikasi" atau "deteksi"), representasi TF-IDF menjadi mirip antar kelas sehingga model lebih mudah tertukar antara RPLD dan SC. Sementara itu, pada kelas SKKKD, istilah terkait jaringan dan keamanan terkadang muncul bersama istilah seperti "anomali" atau "deteksi" yang juga sering digunakan pada SC. Kondisi ini dapat menjelaskan mengapa masih terdapat sebagian prediksi yang keliru pada kelas minor. Dengan demikian, analisis *top terms* mendukung temuan bahwa kesalahan model terutama disebabkan oleh kemiripan istilah lintas bidang serta keterbatasan informasi yang terkandung pada judul yang relatif singkat.

Secara keseluruhan, implementasi sistem ini menggabungkan fungsionalitas klasifikasi, temu kembali judul mirip, dan rekomendasi dosen ke dalam satu aplikasi yang ringan dan mudah dioperasikan. Dengan alur yang terintegrasi, proses kurasi judul menjadi lebih cepat, konsisten, dan terdokumentasi dengan baik sehingga mendukung kebutuhan operasional program studi dalam pengelolaan Tugas Akhir.

3.6 Keterbatasan dan Rencana Pengembangan

Penelitian ini memiliki beberapa keterbatasan yang perlu diperhatikan sebelum hasilnya diaplikasikan secara lebih luas. Pertama, korpus yang digunakan masih terbatas pada judul Tugas Akhir dari satu program studi dan hanya mencakup tiga bidang kajian utama, yaitu RPLD, SC, dan SKKKD. Kondisi ini membuat model sangat terikat pada pola penamaan judul dan cara pelabelan yang berlaku di lingkungan tersebut. Generalisasi ke program studi lain, atau ke skema pengelompokan bidang kajian yang berbeda, belum dapat dijamin tanpa proses penyesuaian dan pelatihan ulang model.

Keterbatasan kedua berkaitan dengan distribusi data yang tidak seimbang, terutama pada kelas SKKKD yang jumlah judulnya jauh lebih sedikit dibandingkan dua kelas lainnya. Walaupun nilai *recall* untuk kelas ini cukup tinggi, variasi pola judul yang dapat dipelajari model tetap terbatas sehingga risiko salah positif masih relatif besar. Di sisi lain, sifat judul sebagai *short text* yang sangat ringkas juga menjadi tantangan tersendiri. Judul yang hanya memuat istilah-istilah generik tanpa konteks tambahan menghasilkan representasi TF-IDF yang kurang kaya, sehingga batas antar kelas menjadi kurang tegas, terutama pada topik-topik lintas bidang.

This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 3, December 2025 Page: 1963−1973
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i3.8859

Dari sisi pendekatan, model yang digunakan masih mengandalkan representasi leksikal berbasis TF-IDF *n-gram* dengan pengklasifikasi *Nearest Centroid*. Pendekatan ini sengaja dipilih karena ringan, sederhana, dan mudah dijelaskan, namun belum memanfaatkan informasi semantik yang lebih dalam seperti yang tersedia pada model representasi teks modern. Akibatnya, judul dengan struktur kalimat yang berbeda tetapi makna serupa bisa saja tidak tertangkap sebagai kedekatan yang kuat di ruang vektor.

Berdasarkan keterbatasan tersebut, terdapat beberapa arah pengembangan yang dapat dilakukan pada penelitian selanjutnya. Dari sisi data, korpus dapat diperluas dengan menambah periode semester, memperkaya contoh judul untuk kelas minor, serta memasukkan program studi lain yang memiliki struktur bidang kajian serupa. Dari sisi fitur, representasi TF-IDF dapat dikombinasikan dengan n-gram karakter atau fitur tambahan yang lebih peka terhadap istilah teknis dan variasi penulisan. Dari sisi model, dapat dieksplorasi pendekatan hibrida yang tetap menjaga interpretabilitas, misalnya dengan tetap menampilkan istilah dominan dan judul tetangga terdekat sebagai penjelasan keputusan model.

Selain itu, integrasi sistem dengan basis data akademik yang sudah ada juga menjadi peluang pengembangan yang menarik. Dengan integrasi yang lebih erat, sistem tidak hanya berfungsi untuk membantu klasifikasi judul dan rekomendasi dosen, tetapi juga dapat dimanfaatkan untuk memantau sebaran topik Tugas Akhir dari waktu ke waktu, mengontrol duplikasi judul, serta mendukung perencanaan kurikulum berbasis data. Dengan serangkaian pengembangan tersebut, diharapkan sistem yang dibangun menjadi semakin matang, dapat direplikasi di lingkungan lain, dan memberikan manfaat yang lebih luas bagi pengelolaan Tugas Akhir di perguruan tinggi.

4. KESIMPULAN

Penelitian ini bertujuan mendukung proses kurasi judul Tugas Akhir pada program studi Teknik Informatika. Fokus utamanya adalah membangun model klasifikasi bidang kajian menggunakan TF-IDF n-gram dan algoritma *Nearest Centroid*, kemudian mengintegrasikannya ke dalam aplikasi *web* yang menyediakan fitur pencarian judul yang mirip serta rekomendasi dosen pembimbing. Evaluasi dilakukan dengan *stratified 5-fold cross-validation* pada 1.057 judul yang telah melalui tahap prapemrosesan dan penyeragaman label. Hasil pengujian menunjukkan akurasi rata rata sekitar 92 persen, dengan nilai F1 yang tinggi pada kelas mayor yaitu RPLD dan SC, serta nilai F1 yang masih memadai pada kelas minor yaitu SKKKD. Temuan ini menunjukkan bahwa model cukup efektif untuk mengelompokkan judul ke tiga bidang kajian utama yang digunakan di program studi. Dari sisi penerapan, sistem menggabungkan klasifikasi, pengecekan kemiripan, dan pemetaan dosen dalam satu alur yang konsisten dan mudah dijalankan kembali. Kondisi ini membantu mempercepat kurasi, meningkatkan konsistensi pengambilan keputusan, dan mendukung proses penelusuran keputusan melalui tampilan *top terms* dan daftar judul terdekat. Sistem juga membantu mahasiswa mengidentifikasi calon pembimbing lebih cepat dan mengurangi beban peninjauan manual bagi koordinator. Adapun keterbatasan penelitian terutama terkait ketidakseimbangan jumlah data antar kelas, khususnya pada kelas minor, serta karakter judul yang cenderung ringkas. Penelitian selanjutnya dapat diarahkan pada perluasan korpus, strategi penyeimbangan data, penambahan n-gram karakter, serta eksplorasi representasi hibrida yang tetap ringan agar ketahanan pada kasus batas meningkat tanpa mengurangi keterjelasan dan efisiensi komputasi.

REFERENCES

[1] S. Chawla, R. Kaur, and P. Aggarwal, “Text classification framework for short text based on TFIDF-FastText,” *Multimed Tools Appl*, vol. 82, no. 26, pp. 40167–40180, Nov. 2023, doi: 10.1007/s11042-023-15211-5.

[2] Z. Khan, U. Naseer, and M. A. Tahir, “Short Text Classification using TF-IDF Features and FastText Learner,” in *Working Notes Proceedings of the MediaEval 2021 Workshop*, 2021. [Online]. Available: https://ceur-ws.org/Vol-3181/paper59.pdf

[3] A. D. D. Wibiyanto and A. Wibowo, “PENERAPAN ALGORITMA MULTICLASS SUPPORT VECTOR MACHINE DAN TF-IDF UNTUK KLASIFIKASI TOPIK TUGAS AKHIR,” *SKANIKA*, vol. 6, no. 1, pp. 42–50, Jan. 2023, doi: 10.36080/skanika.v6i1.2999.

[4] A. H. Nasrullah, “Integrasi TF-IDF dan Algoritma Cosine Similarity untuk Deteksi Tingkat Kemiripan Judul Tugas Akhir,” *INTEC Journal: Information Technology and Education*, vol. 4, no. 1, pp. 1–10, 2024, Accessed: Dec. 15, 2025. [Online]. Available: https://journal.unm.ac.id/index.php/INTEC/article/view/5810

[5] D. Meidelfi, - Yulherniwati, I. Rahmayuni, T. Hidayat, and D. Chandra, “TF-IDF Implementation for Similarity Checker on The Final Project Title,” *International Journal of Advanced Science Computing and Engineering*, vol. 3, no. 1, pp. 40–52, Oct. 2021, doi: 10.62527/ijasce.3.1.3.

[6] R. Ardianzah and H. Thamrin, “Pengembangan Sistem Pencarian Pada Aplikasi Skripsi Untuk Meningkatkan Hasil Pencarian Judul,” 2024. [Online]. Available: https://eprints.ums.ac.id/120772/1/Naskah%20Publikasi.pdf

[7] F. D. Astuti and W. Andriyani, “Pengembangan Sistem Rekomendasi Pembimbing Tugas Akhir Menggunakan Teknik Content Based Filtering,” *JIKO (Jurnal Informatika dan Komputer)*, vol. 9, no. 2, p. 474, Jun. 2025, doi: 10.26798/jiko.v9i2.1599.

[8] I. Mawanta, T. S. Gunawan, and W. Wanayumini, “Uji Kemiripan Kalimat Judul Tugas Akhir dengan Metode Cosine Similarity dan Pembobotan TF-IDF,” *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 5, no. 2, p. 726, Apr. 2021, doi: 10.30865/mib.v5i2.2935.

[9] Ritzkal, W. T. Atmojo, P. Novantara, S. Rosidin, A. D. Jubaedi, and E. Novianto, “Improving Thesis Title Classification Accuracy Using Ensemble Classifier and Modified Chi-Square Feature Selection Method,” *Indonesian Applied Research on*

Copyright © 2025 **Author**, Page 1972
This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 3, December 2025 Page: 1963−1973
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i3.8859

Computing and Informatics, vol. 1, no. 1, pp. 37–47, 2025, Accessed: Dec. 15, 2025. [Online]. Available: https://jurnal.tdinus.com/index.php/iarci/article/view/52

[10] J.-W. Sun, J.-Q. Bao, and L.-P. Bu, "Text Classification Algorithm Based on TF-IDF and BERT," in *2022 11th International Conference of Information and Communication Technology (ICTech))*, IEEE, Feb. 2022, pp. 1–4. doi: 10.1109/ICTech55460.2022.00112.

[11] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP," in *Proceedings of the 28th International Conference on Computational Linguistics*, Stroudsburg, PA, USA: International Committee on Computational Linguistics, 2020, pp. 757–770. doi: 10.18653/v1/2020.coling-main.66.

[12] Sutriawan, S. Rustad, G. F. Shidik, and Pujiono, "Performance Evaluation of Text Embedding Models for Ambiguity Classification in Indonesian News Corpus: A Comparative Study of TF-IDF, Word2Vec, FastText BERT, and GPT," *Ingénierie des systèmes d information*, vol. 30, no. 6, pp. 1469–1482, Jun. 2025, doi: 10.18280/isi.300606.

[13] M. Liang and T. Niu, "Research on Text Classification Techniques Based on Improved TF-IDF Algorithm and LSTM Inputs," *Procedia Comput Sci*, vol. 208, pp. 460–470, 2022, doi: 10.1016/j.procs.2022.10.064.

[14] C. Li, Z. Xie, and H. Wang, "Short Text Classification Based on Enhanced Word Embedding and Hybrid Neural Networks," *Applied Sciences*, vol. 15, no. 9, p. 5102, May 2025, doi: 10.3390/app15095102.

[15] P. Sayarizki, Hasmawanti, and H. Nurrahmi, "Implementation of IndoBERT for Sentiment Analysis of Indonesian Presidential Candidates," *Indonesian Journal of Computing*, vol. 9, no. 2, pp. 1–11, 2024, doi: 10.34818/INDOJC.2024.9.2.934.

[16] D. E. Cahyani and I. Patasik, "Performance comparison of TF-IDF and Word2Vec models for emotion text classification," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 5, pp. 2780–2788, Oct. 2021, doi: 10.11591/eei.v10i5.3157.

[17] E. Yuniar and N. Hendrastuty, "Perbandingan Metode Naive Bayes, Random Forest dan SVM untuk Analisis Sentimen pada Twitter tentang Kenaikan Gaji Guru," *Building of Informatics, Technology and Science (BITS)*, vol. 6, no. 4, pp. 2469–2479, 2025, doi: 10.47065/bits.v6i4.6970.

[18] Dwi Nanda Agustia and Ryan Randy Suryono, "Comparison of Naïve Bayes, Random Forest, and Logistic Regression Algorithms for Sentiment Analysis Online Gambling," *INOVTEK Polbeng - Seri Informatika*, vol. 10, no. 1, pp. 284–295, Jan. 2025, doi: 10.35314/prk93630.

[19] C. C. Aggarwal, *Machine Learning for Text*. Cham: Springer International Publishing, 2018. doi: 10.1007/978-3-319-73531-3.

[20] S. Raschka, Y. (Hayden) Liu, and V. Mirjalili, *Machine Learning with PyTorch and Scikit-Learn: Develop Machine Learning and Deep Learning Models with Python*. Birmingham: Packt Publishing, 2022.

[21] J. Albrecht, S. Ramachandran, and C. Winkler, *Blueprints for Text Analytics Using Python: Machine Learning-Based Solutions for Common Real World (NLP) Applications*. Sebastopol, CA: O'Reilly Media, 2021.

[22] I. Nyoman Prayana Trisna, N. Wayan Emmy Rosiana Dewi, and M. Alam Pasirulloh, "Oversampling vs. undersampling in TF-IDF variations for imbalanced Indonesian short texts classification," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 23, no. 2, p. 382, Apr. 2025, doi: 10.12928/telkomnika.v23i2.26510.

[23] B. Trstenjak, S. Mikac, and D. Donko, "KNN with TF-IDF based Framework for Text Categorization," *Procedia Eng*, vol. 69, pp. 1356–1364, 2014, doi: 10.1016/j.proeng.2014.03.129.

[24] N. Arifin, U. Enri, and N. Sulistiyowati, "Penerapan Algoritma Support Vector Machine (SVM) dengan TF-IDF N-Gram untuk Text Classification," *STRING (Satuan Tulisan Riset dan Inovasi Teknologi)*, vol. 6, no. 2, p. 129, Dec. 2021, doi: 10.30998/string.v6i2.10133.

[25] I. R. Illahi and E. B. Setiawan, "Sentiment Analysis on Social Media Using Fasttext Feature Expansion and Recurrent Neural Network (RNN) with Genetic Algorithm Optimization," *International Journal on Information and Communication Technology (IJoICT)*, vol. 10, no. 1, pp. 78–89, Jun. 2024, doi: 10.21108/ijoict.v10i1.905.

[26] D. R. Firmansyah and E. Lestariningsih, "Analisis Sentimen Ulasan Aplikasi Smart Campus Unisbank di Google Playstore Menggunakan Algoritma Naive Bayes," *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)*, vol. 8, no. 2, pp. 498–507, Apr. 2024, doi: 10.35870/jtik.v8i2.1882.

This Journal is licensed under a Creative Commons Attribution 4.0 International License