Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi
https://ojs.stmik-banjarbaru.ac.id/index.php/jutisi/index
Jl. Ahmad Yani, K.M. 33,5 - Kampus STMIK Banjarbaru
Loktabat – Banjarbaru (Tlp. 0511 4782881), e-mail: puslit.stmikbjb@gmail.com
e-ISSN: 2685-0893

Evaluasi Performa Algoritma Klasterisasi dalam Mengelompokkan Topik Tugas Akhir: Studi Komparasi *K-Means* dan DBSCAN

DOI: http://dx.doi.org/10.35889/jutisi.v15i3.3759

Tifanny Nabarian[1*], Dhea Marsella[2], Salman El Farisi[3]
Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri, Depok, Indonesia
*e-mail *Corresponding Author*: dheamarsella43@gmail.com

Abstrak

The increasing number of Informatics Engineering students at STT-NF has resulted in a growing volume of final project topics that need to be efficiently categorized into research areas. This study evaluates the performance of the K-Means and DBSCAN algorithms for clustering students' final project topics based on their textual characteristics. The research follows the CRISP-DM framework and utilizes a dataset of 656 final project titles. The data were processed through text preprocessing, TF-IDF weighting, and dimensionality reduction PCA. The experimental results indicate that K-Means with k = 13 achieved the best performance, obtaining a silhouette score of 0.1434 and a purity score of 0.8659, outperforming DBSCAN, which achieved a silhouette score of 0.1153 and a purity score of 0.6589. The resulting clusters were successfully mapped into four major research areas, namely Software Engineering, Network Engineering & Cyber Security, Data Engineering, and UI/UX Design. Furthermore, the best-performing model was implemented in an interactive Streamlit-based dashboard to support research area mapping and academic supervisor assignment.

Keywords: *K-Means; DBSCAN; Final Project Topic Clustering; TF-IDF*

Abstrak

Peningkatan jumlah mahasiswa Teknik Informatika di STT-NF menghasilkan semakin banyak data topik tugas akhir yang perlu dikelompokkan ke dalam bidang penelitian secara efisien. Penelitian ini mengevaluasi performa algoritma K-Means dan DBSCAN untuk mengelompokkan topik tugas akhir mahasiswa berdasarkan kemiripan karakteristiknya. Proses penelitian mengikuti metodologi CRISP-DM dengan menggunakan 656 judul tugas akhir yang diproses melalui tahapan pemrosesan teks, pembobotan TF-IDF, dan reduksi dimensi PCA. Berdasarkan hasil pengujian, K-Means dengan k = 13 memberikan kinerja terbaik dengan nilai silhouette score 0.1434 dan *purity* 0.8659, sedangkan DBSCAN memperoleh nilai *silhouette score* 0.1153 dan *purity* 0.6589. Klaster yang terbentuk berhasil direpresentasikan ke dalam empat bidang penelitian utama, yaitu *Software Engineering*, *Network Engineering & Cyber Security*, *Data Engineering*, dan *UI/UX Design*. Sebagai implementasi, model terbaik diintegrasikan ke dalam *dashboard* interaktif berbasis Streamlit untuk mendukung pemetaan bidang penelitian dan penentuan dosen pembimbing.

Kata kunci: *K-Means; DBSCAN; Klasterisasi Topik Tugas Akhir; TF-IDF*

1. Pendahuluan

Perkembangan teknologi informasi serta meningkatnya jumlah mahasiswa pada program studi informatika menyebabkan semakin banyak karya ilmiah yang dihasilkan dalam bentuk tugas akhir. Tugas akhir tidak hanya menjadi bentuk evaluasi akademik mahasiswa, tetapi juga mencerminkan perkembangan topik penelitian dalam suatu program studi. Seiring dengan meningkatnya jumlah mahasiswa, variasi topik penelitian yang dihasilkan juga semakin beragam sehingga menimbulkan tantangan dalam proses pengelompokan dan pemetaan bidang penelitian secara sistematis.

Program Studi Teknik Informatika Sekolah Tinggi Teknologi Terpadu Nurul Fikri (STT-NF) setiap tahunnya menghasilkan ratusan topik tugas akhir dengan berbagai fokus penelitian seperti *Software Engineering, Network Engineering, Cyber Security, Data Engineering,* dan *UI/UX Design*. Keberagaman topik tersebut dapat menyulitkan proses identifikasi kesamaan tema penelitian apabila dilakukan secara manual. Selain itu, proses distribusi topik tugas akhir kepada dosen pembimbing yang memiliki bidang keahlian relevan juga menjadi kurang optimal tanpa adanya dukungan analisis data yang terstruktur. Kondisi tersebut memerlukan pendekatan berbasis pengolahan data yang mampu mengelompokkan topik penelitian secara otomatis melalui kesamaan karakteristik teks.

Penelitian ini memanfaatkan teknik *text mining* untuk mengolah dan menganalisis data topik tugas akhir dalam bentuk teks tidak terstruktur [1]. Dalam pengolahan teks, dokumen umumnya direpresentasikan ke dalam bentuk numerik menggunakan metode TF-IDF. Melalui metode ini, setiap kata memperoleh bobot yang mencerminkan kontribusinya terhadap isi dokumen serta tingkat penyebarannya pada seluruh korpus, sehingga kata-kata yang lebih representatif dapat diidentifikasi [2]. Setelah data teks direpresentasikan dalam bentuk numerik, proses pengelompokan dapat dilakukan menggunakan teknik klasterisasi. Klasterisasi merupakan proses membagi sekumpulan data ke dalam beberapa kelompok berdasarkan tingkat kemiripan antarobjek. Data yang berada dalam kelompok yang sama diharapkan memiliki karakteristik serupa dibandingkan dengan data pada kelompok lainnya [3]. Algoritma yang banyak diterapkan dalam klasterisasi dokumen teks antara lain K-Means dan DBSCAN. K-Means melakukan pengelompokan dengan menempatkan data pada klaster yang memiliki *centroid* terdekat, sedangkan DBSCAN membangun klaster berdasarkan tingkat kepadatan antar data. Data yang berada di luar area kepadatan yang memadai akan ditandai sebagai *noise* atau pencilan [4].

Sejumlah penelitian terdahulu telah menerapkan klastersasi pada dokumen teks, khususnya pada topik tugas akhir. Penelitian [5] menerapkan kombinasi text mining dan K-Means untuk pengelompokan dokumen skripsi dengan memperoleh nilai silhouette Score terbaik sebesar 0.4837 pada k = 4. Penelitian [6] mengelompokkan topik penelitian berbasis dokumen tugas akhir dan tesis menggunakan K-Means, TF-IDF, serta PCA dengan nilai silhouette coefficient terbaik sebesar 0.7119. Sementara itu penelitian [7], menerapkan DBSCAN pada klasterisasi topik skripsi informatika dan menunjukkan bahwa kombinasi TF-IDF, reduksi dimensi, serta pemilihan parameter yang tepat mampu menghasilkan kualitas klaster yang baik. Penelitian lain juga menunjukkan bahwa *K-Means* dapat dimanfaatkan untuk mengelompokkan dokumen tugas akhir maupun mendukung pemetaan topik skripsi mahasiswa [8], [9]. Berdasarkan tinjauan tersebut, sebagian besar penelitian klasterisasi topik tugas akhir masih menggunakan *K-Means* sebagai metode utama, sedangkan penelitian yang menerapkan DBSCAN umumnya dilakukan secara terpisah. Selain itu, perbandingan langsung antara algoritma K-Means dan DBSCAN pada data topik tugas akhir berbahasa Indonesia masih relatif terbatas. Akibatnya, belum diketahui secara jelas algoritma yang lebih sesuai untuk menghasilkan klaster yang representatif pada konteks pengelolaan topik tugas akhir mahasiswa.

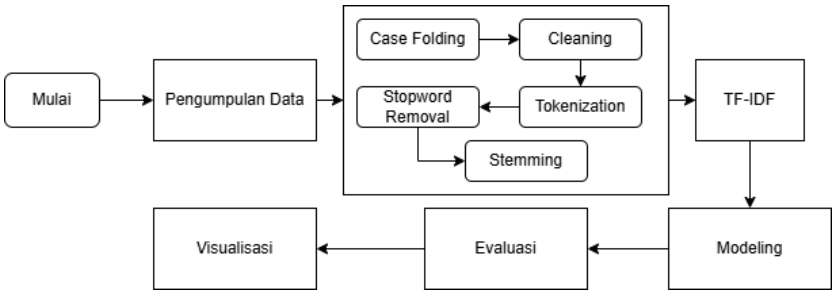
Sebagai Solusi terhadap permasalahan tersebut, penelitian ini menerapkan klasterisasi berbasis teks dengan representasi fitur menggunakan TF-IDF. Kinerja algoritma K-Means dan DBSCAN kemudian dibandingkan melalui evaluasi menggunakan silhouette coefficient dan *purity*. Pemilihan TF-IDF didasarkan pada efektivitasnya dalam mengekstrak fitur teks pendek akademik dalam memberikan bobot tinggi pada kata unik dan menekan bobot kata umum [10]. Pemilihan K-Means dan DBSCAN didasarkan pada karakteristik keduanya yang berbeda dalam membentuk klaster. K-Means mengelompokkan data berdasarkan kedekatan terhadap *centroid* dan telah banyak digunakan pada penelitian klasterisasi dokumen tugas akhir dengan performa yang baik [5], [6], sedangkan DBSCAN memiliki kemampuan mengidentifikasi *noise* serta membentuk klaster tanpa menentukan jumlah di awal, sehingga relevan digunakan pada data topik tugas akhir yang memiliki keragaman tema dan struktur kelompok yang belum diketahui secara pasti [7]. Seluruh proses penelitian mengacu pada metodologi CRISP-DM sebagai kerangka kerja analisis data yang sistematis [11].

Berdasarkan beberapa penelitian terdahulu yang umumnya berfokus pada satu algoritma, penelitian ini mengevaluasi kedua metode secara komparatif. Komparasi ini didukung oleh studi literatur yang menunjukkan bahwa karakter teks pendek memiliki *sparsity* data yang tinggi, di mana performa berbasis densitas dan *centroid* sering kali menunjukkan kontras yang

signifikan tergantung persebaran datanya [12]. *State of the art* penelitian ini terletak pada evaluasi perbandingan algoritma K-Means dan DBSCAN pada data topik tugas akhir berbahasa Indonesia untuk mengidentifikasi metode yang paling sesuai dalam menghasilkan klaster yang representatif, serta mengimplementasikan hasil klasterisasi dalam bentuk *dashboard* interaktif untuk mendukung pengambilan keputusan akademik.

2. Metodologi

Penelitian ini menggunakan pendekatan *Cross-Industry Standard Process for Data Mining* (CRISP-DM) sebagai kerangka kerja dalam proses analisis data. Kerangka kerja CRISP-DM digunakan dalam penelitian ini karena menyediakan tahapan yang terstruktur untuk mengarahkan proses analisis data. Tahapan yang diterapkan mencakup *Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation,* dan *Deployment*. Rangkaian proses penelitian tersebut dirangkum pada Gambar 1.



Gambar 1. Tahapan Penelitian

2.1 *Business Understanding*

Pada tahap *Business Understanding*, fokus utama penelitian adalah menganalisis kebutuhan terkait pengelompokan topik tugas akhir yang selama ini masih bergantung pada proses manual. Untuk membantu proses tersebut, penelitian ini menerapkan pendekatan klasterisasi berbasis teks guna menemukan pola keterkaitan antar topik dan mengelompokkannya ke dalam bidang penelitian yang relevan. Performa algoritma K-Means dan DBSCAN kemudian dievaluasi untuk menentukan metode yang paling sesuai dalam menghasilkan klaster yang representatif. Hasil yang diperoleh diharapkan dapat memberikan informasi mengenai persebaran topik tugas akhir serta menjadi bahan pendukung dalam pemetaan bidang penelitian di lingkungan akademik.

2.2 *Data Understanding*

Tahap ini dilakukan guna mengidentifikasi karakteristik data penelitian. Dataset berupa kumpulan judul tugas akhir mahasiswa Prodi Teknik Informatika STT-NF pada periode 2023-2024 dengan total 657 dokumen. Setiap data merupakan teks pendek (*short text*) yang merepresentasikan topik penelitian mahasiswa. Karakteristik teks yang relatif singkat dengan variasi kata yang tinggi menyebabkan data memerlukan tahapan praproses sebelum dilakukan proses klasterisasi. Tabel 1 di bawah ini memberikan gambaran umum mengenai dataset yang digunakan dalam penelitian.

Tabel 1. Dataset Topik TA

No	Topik TA	Bidang Penelitian
1	Analisis dan Perancangan Representational Rest Transfer Web Service Sistem Informasi Akademik STT Terpadu Nurul Fikri Menggunakan Yii Framework	NaN
2	Perancangan Sistem Informasi Keuangan Berbasis Mobile	NaN
3	Penerapan Model Role Base Acces Control Pada Pengembangan Sistem Informasi Akademik STT-NF Berbasis Web Menggunakan XII Framework	NaN

Kolom Topik Tugas Akhir berisi teks yang merepresentasikan topik penelitian mahasiswa, sedangkan bidang penelitian ditentukan berdasarkan hasil klasterisasi. Penentuan bidang dilakukan melalui identifikasi *top terms* pada setiap klaster yang merepresentasikan karakteristik utama kelompok data. Sebagai contoh, klaster yang didominasi istilah terkait perancangan dan pengembangan sistem dikategorikan ke dalam bidang *Software Engineering*.

2.3 *Data Preparation*

Tahap *data preparation* dilakukan untuk mempersiapkan dataset sebelum proses klasterisasi. Tahapan ini meliputi *preprocessing* teks, ekstraksi fitur menggunakan metode *Term Frequency–Inverse Document Frequency* (TF-IDF), serta reduksi dimensi menggunakan *Principal Component Analysis* (PCA). *Preprocessing* dilakukan untuk mempersiapkan data teks melalui serangkaian proses pembersihan dan penyeragaman data agar lebih siap untuk dianalisis pada tahapan berikutnya. Tahapan *preprocessing* yang dilakukan meliputi:

1) *Case Folding*
   Case folding merupakan proses mengubah huruf kapital pada sebuah kata menjadi huruf kecil (*lowercase).* Proses ini mencegah kata yang memiliki makna sama, seperti "Study" dan "study", dianggap sebagai entitas yang berbeda [13].
2) *Cleaning*
   Cleaning merupakan proses pembersihan teks dengan menghapus karakter khusus, angka, URL, serta simbol yang tidak relevan sehingga tidak memengaruhi proses pembobotan kata dan hasil klasterisasi [14].
3) *Tokenization*
   Tahap tokenization dilakukan dengan membagi teks menjadi sekumpulan token yang merepresentasikan kata, frasa, maupun simbol tertentu sebagai unit dasar pengolahan teks [13].
4) *Stopword Removal*
   Tahap ini dilakukan untuk menyaring berbagai kata yang memiliki frekuensi kemunculan tinggi namun kontribusi informasinya rendah terhadap isi dokumen. Kata-kata tersebut dihapus karena kemunculannya sangat umum dan tidak memberikan informasi penting dalam analisis teks [14].
5) *Stemming*
   Proses menyederhanakan kata berimbuhan menjadi kata dasar (*root word*) dengan cara menghilangkan afiksasi. Dengan proses ini, kata yang memiliki akar yang sama seperti “pelajar” dan “belajar” dapat dikembalikan ke bentuk dasar “ajar” [14].

Setelah melalui tahapan *preprocessing*, setiap dokumen diubah ke dalam bentuk fitur numerik menggunakan metode TF-IDF. Metode ini digunakan untuk mengekstraksi informasi penting dari teks dengan memberikan nilai bobot yang berbeda pada setiap kata sesuai tingkat kontribusinya terhadap isi dokumen dan distribusinya dalam korpus. Dengan pendekatan tersebut, istilah yang lebih mampu merepresentasikan karakteristik suatu dokumen akan memperoleh bobot yang lebih tinggi dibandingkan kata-kata yang muncul secara umum pada banyak dokumen [15]. Perhitungan TF-IDF secara umum ditunjukkan pada persamaan berikut.

$$W_{dt} = TF_{dt} * IDF_t \quad (1)$$

$$IDF_t = \log (N/df_t) \quad (2)$$

Setelah proses pembobotan kata menggunakan TF-IDF, fitur yang dihasilkan memiliki dimensi yang tinggi sehingga berpotensi menimbulkan permasalahan *curse of dimensionality*. Oleh karena itu, dilakukan reduksi dimensi menggunakan PCA. PCA merupakan teknik statistik yang digunakan untuk mengurangi jumlah fitur dengan tetap mempertahankan informasi utama dalam data melalui transformasi variabel asli menjadi beberapa komponen utama yang merepresentasikan variasi terbesar dalam dataset [16]. Selain itu, penerapan PCA juga bertujuan untuk mengurangi *noise* pada data serta meningkatkan kinerja algoritma klasterisasi, khususnya DBSCAN yang sensitif terhadap kepadatan data. Penerapan reduksi dimensi yang seragam dalam penelitian ini juga bertujuan untuk menjaga konsistensi data masukan, sehingga perbandingan performa antara algoritma K-Means dan DBSCAN dapat dilakukan secara adil.

2.4 *Modeling*

Data hasil praproses kemudian digunakan dalam tahap pemodelan dengan menerapkan algoritma K-Means dan DBSCAN guna mengelompokkan topik tugas akhir berdasarkan kemiripan representasi teks.

1) K-Means

Proses klasterisasi menggunakan K-Means dilakukan dengan menghitung jarak antara setiap data dan pusat klaster (*centroid*) menggunakan *Euclidean Distance* [17]. Algoritma ini bekerja mengelompokan data berdasarkan kedekatannya terhadap *centroid* yang mewakili setiap klaster. Setelah proses pengelompokan, *centroid* diperbarui menggunakan nilai rata-rata anggota klaster dan proses tersebut diulang sampai diperoleh kondisi yang stabil atau tidak terjadi perubahan *centroid* yang berarti.

$$d(x_i, c_j)\sqrt{\sum_{k=1}^{n}(x_{ik} - c_{jk})} \quad (3)$$

2) DBSCAN

DBSCAN mengelompokkan data berdasarkan tingkat kepadatan pada ruang fitur dan mampu mendeteksi data yang menyimpang dari pola umum sebagai *noise* (outlier) [4]. Dalam prosesnya, data dikategorikan menjadi *core point, border point,* dan *noise*. Pembentukan klaster ditentukan oleh parameter ε (epsilon) yang mengatur radius pencarian tetangga serta MinPts yang menyatakan jumlah minimum titik untuk membentuk sebuah klaster, dengan perhitungan jarak menggunakan Euclidean Distance [17].

$$d_{ij}\sqrt{\sum(x_{ia} - x_{ja})^2} \quad (4)$$

Eksperimen dilakukan dengan variasi jumlah (k) pada K-Means serta beberapa kombinasi nilai ε dan *MinPts* pada DBSCAN. Parameter optimal dipilih berdasarkan nilai evaluasi terbaik.

2.5 *Evaluation*

Evaluasi kualitas klaster pada penelitian ini dilakukan menggunakan *silhouette coefficient*. Metrik tersebut menghasilkan nilai dalam rentang −1 hingga 1, yang mencerminkan kualitas struktur klaster yang terbentuk. Nilai yang lebih besar menunjukkan bahwa data dalam klaster memiliki tingkat kemiripan tinggi dan terpisah dengan baik dari klaster lain. Sebaliknya, nilai yang mendekati nol atau bernilai negatif mengindikasikan bahwa batas antar klaster kurang jelas. Perhitungan silhouette didasarkan pada perbandingan kedekatan data terhadap anggota dalam klasternya sendiri dan terhadap klaster terdekat. Nilai silhouette untuk setiap data dihitung menggunakan persamaan berikut [18].

$$\mathrm{s(i)}\frac{\mathrm{b(i)} - a(i)}{\max(a(i), (b(i))} \quad (5)$$

Selain metrik internal, evaluasi klaster juga dilakukan menggunakan metrik eksternal, yaitu *purity*. *Purity* mengukur kesesuaian antara hasil klasterisasi dan label sebenarnya dengan menghitung proporsi anggota mayoritas pada setiap klaster terhadap total data. Nilai *purity* yang lebih tinggi menunjukkan kualitas klaster yang lebih baik [10]. Nilai *purity* dihitung menggunakan persamaan berikut:

$$purity = \frac{jumlah\ label\ terbanyak\ tiap\ klaster}{total\ dokumen} \quad (6)$$

Hasil evaluasi dari kedua algoritma kemudian dianalisis secara komparatif untuk mengidentifikasi perbedaan performa dalam menghasilkan klaster yang berkualitas, sehingga dapat ditentukan metode yang lebih efektif dalam mengelompokkan data sesuai karakteristik yang dimiliki.

2.6 *Deployment*

Hasil penelitian diimplementasikan dalam bentuk *dashboard* interaktif menggunakan Streamlit. *Dashboard* ini dirancang untuk memfasilitasi prediksi topik tugas akhir berdasarkan bidang penelitian dengan memanfaatkan model klasterisasi yang telah dibangun. Selain itu, *dashboard* juga menyajikan visualisasi hasil klasterisasi serta distribusi data untuk mendukung interpretasi hasil. Pengembangan *dashboard* ini bertujuan untuk mempermudah pengguna dalam mengidentifikasi kecenderungan topik tugas akhir sesuai dengan bidang yang dipilih.

3. Hasil dan Pembahasan

Bagian ini menyajikan hasil eksperimen dan pembahasan dari penerapan metode klasterisasi pada data topik tugas akhir. Hasil yang diperoleh dianalisis untuk mengevaluasi kualitas klaster serta mengidentifikasi pola yang terbentuk. Seluruh proses dalam penelitian ini mengacu pada metodologi CRISP-DM.

3.1 Data Understanding

Data understanding bertujuan mengidentifikasi karakteristik dataset sebagai dasar dalam proses pengolahan dan analisis selanjutnya. Berdasarkan distribusi data pada setiap periode, jumlah topik tugas akhir menunjukkan variasi yang cukup signifikan. Periode 2023-1 memiliki jumlah data terbanyak, yaitu 304 topik (46,3%) dari total data. Sementara itu, periode 2023-2 memiliki jumlah 109 topik (16,6%), diikuti periode 2024-1 sebanyak 78 topik (11,9%), dan periode 2024-2 sebanyak 166 topik (25,2%). Secara keseluruhan, total data yang dianalisis berjumlah 657 topik, yang menunjukkan bahwa distribusi data antar periode tidak merata. Selain itu, berdasarkan analisis jumlah kata pada setiap topik, diperoleh rata-rata panjang teks sebesar 14 kata. Panjang teks terpendek terdiri dari 3 kata, sedangkan yang terpanjang mencapai 26 kata. Secara umum, sebagian besar topik memiliki panjang teks yang relatif pendek dan tidak berbeda jauh satu sama lain.

3.2 *Data Preparation*

Tahap ini merupakan fondasi penting dalam *text mining* karena memengaruhi performa algoritma klasterisasi. Data teks mentah umumnya tidak terstruktur dan mengandung elemen yang tidak relevan, sehingga perlu diproses sebelum dianalisis. Sebelum memasuki tahap *preprocessing*, dilakukan pengecekan terhadap data kosong (*null*) dan duplikasi data. Hasil pengecekan menunjukkan ditemukan 1 data duplikat sehingga jumlah data berkurang dari 657 menjadi 656. Selanjutnya, perubahan data pada setiap tahapan *preprocessing* disajikan di Tabel 2.

Tabel 2. Hasil *Preprocessing*

Tahapan	Topik TA
Topik Awal	Analisis Perancangan Sistem Informasi Akademik Modul Kartu Hasil Studi Berbasis mobile dengan Pendekatan Native dan Hybrid Kasus STT Terpadu Nurul Fikri dengan Aspek Pengujian ISO 25010
Case Folding	analisis perancangan sistem informasi akademik modul kartu hasil studi berbasis mobile dengan pendekatan native dan hybrid kasus stt terpadu nurul fikri dengan aspek pengujian iso 25010
Cleaning	analisis perancangan sistem informasi akademik modul kartu hasil studi berbasis mobile dengan pendekatan native dan hybrid kasus stt terpadu nurul fikri dengan aspek pengujian iso
Tokenization	[analisis, perancangan, sistem, informasi, akademik, modul, kartu, hasil, studi, berbasis, mobile, dengan, pendekatan, native, dan, hybrid, kasus, stt, terpadu, nurul, fikri, dengan, aspek, pengujian, iso]
Stopword Removal	[analisis, perancangan, sistem, informasi, akademik, modul, kartu, hasil, mobile, pendekatan, native, hybrid, aspek, pengujian, iso]
Stemming	[analisis, rancang, sistem, informasi, akademik, modul, kartu, hasil, mobile, dekat, native, hybrid, aspek, uji, iso]
Hasil Pemroses	analisis rancang sistem informasi akademik modul kartu hasil mobile dekat native hybrid aspek uji iso

Berdasarkan hasil *preprocessing*, setiap tahapan memberikan kontribusi reduksi yang berbeda terhadap data. *Stopword removal* menjadi tahap paling signifikan, ditunjukkan oleh penurunan jumlah karakter dari 71.756 menjadi 55.608 serta total token dari 9.360 menjadi 7.088. Hal ini menunjukkan bahwa sebagian besar kata dalam dataset merupakan kata umum yang kurang membedakan topik, seperti nama perusahaan, organisasi, dan istilah umum lainnya. Oleh karena itu, *preprocessing* berperan penting dalam meningkatkan kualitas representasi data sebelum klasterisasi.

Setelah *preprocessing*, data teks diubah ke dalam bentuk numerik menggunakan metode TF-IDF. Jumlah fitur dibatasi hingga 500 menggunakan max_features untuk mengurangi dimensi data, menekan *sparsity*, dan meningkatkan efisiensi komputasi. Parameter ngram_range (1,3) digunakan untuk menangkap unigram hingga trigram, sedangkan min_df = 2 dan max_df = 0.9 digunakan untuk menghilangkan term yang terlalu jarang maupun terlalu umum. Selain itu, sublinear_tf diterapkan untuk mentransformasikan frekuensi *term* ke skala logaritmik sehingga mengurangi dominasi *term* dengan frekuensi tinggi. Hasil pembobotan menghasilkan matriks fitur berukuran 656 × 500 yang merepresentasikan 656 dokumen dan 500 fitur. Matriks tersebut disajikan pada Tabel 3.

Tabel 3. Matriks TF-IDF

ID Docx	absensi	ajar	akademik	akademik web	algoritma	…	web metode design
D0	0.0000	0.0000	0.2641	0.0000	0.0000	…	0.0000
D1	0.0000	0.0000	0.0000	0.0000	0.0000	…	0.0000
D2	0.0000	0.0000	0.2892	0.3905	0.0000	…	0.0000
D3	0.0000	0.0000	0.2425	0.0000	0.0000	…	0.0000
…	…	…	…	…	…	…	…
D655	0.0000	0.0000	0.0000	0.0000	0.3209	…	0.0000

Berdasarkan Tabel 3, setiap dokumen direpresentasikan dalam bentuk vektor numerik yang terdiri atas 500 fitur dengan bobot TF-IDF. Sebagian besar nilai pada matriks bernilai nol karena tidak semua term muncul pada setiap dokumen, sehingga menghasilkan matriks yang bersifat *sparse*. Untuk mengurangi kompleksitas data berdimensi tinggi, dilakukan reduksi dimensi menggunakan PCA. Pengujian dilakukan pada beberapa nilai *explained variance* untuk menganalisis pengaruh jumlah komponen terhadap kualitas klaster. Berdasarkan hasil eksperimen, nilai n_components = 0.50 dipilih karena mampu mempertahankan sekitar 50% variasi data dengan 53 komponen utama, sehingga jumlah fitur dapat dikurangi secara signifikan tanpa reduksi informasi yang terlalu besar. Setelah proses PCA, ukuran data berkurang dari 656 × 500 menjadi 656 × 53. Hasil eksperimen PCA ditampilkan pada Tabel 4.

Tabel 4. Eksperiment PCA

PCA Components	Dimensi Akhir	Explained Variance	Silhouette Score
0.2	10	0.2037	0.3064
0.3	21	0.3041	0.1950
0.4	35	0.4011	0.1491
0.5	53	0.5003	0.1178
0.6	76	0.6013	0.0954
0.7	106	0.7026	0.0769
0.8	146	0.8019	0.0659
0.9	207	0.9009	0.0570

Berdasarkan Tabel 4, peningkatan *explained variance* menyebabkan jumlah komponen yang dipertahankan semakin banyak, namun nilai silhouette score cenderung menurun. Meskipun *explained variance* 0.20 menghasilkan silhouette score tertinggi, konfigurasi tersebut hanya mempertahankan sekitar 20% variasi data. Oleh karena itu, penelitian ini memilih *explained variance* 0.50 sebagai kompromi antara kualitas klaster dan informasi yang dipertahankan dari data awal. Selain itu, reduksi dimensi membantu mengurangi dampak *curse of dimensionality* pada data teks berdimensi tinggi sehingga perhitungan jarak menjadi lebih

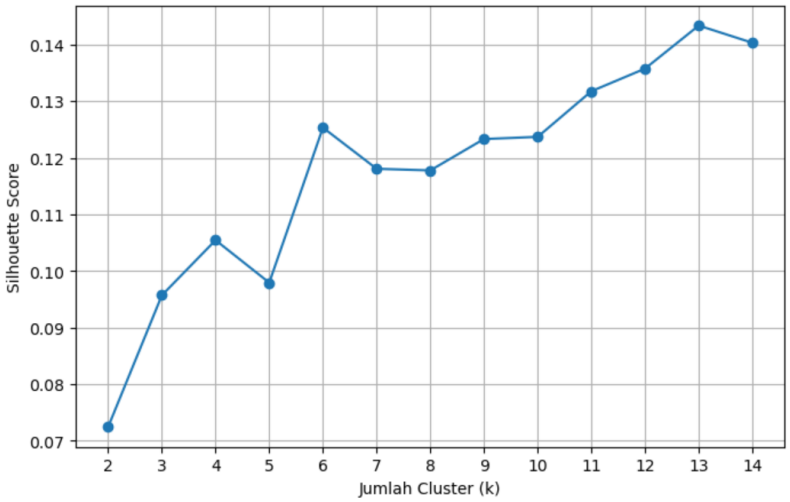
efektif dan *noise* dapat diminimalkan [19]. Temuan ini menunjukkan bahwa mempertahankan komponen yang lebih banyak tidak selalu meningkatkan kualitas klasterisasi.

3.3 *Modeling*

Tahap modeling dilakukan untuk menentukan parameter yang paling sesuai pada algoritma K-Means dan DBSCAN guna menghasilkan kualitas klaster yang optimal. Hasil pengujian parameter pada masing-masing algoritma disajikan pada bagian berikut.

1) K-Means

Pada algoritma K-Means, dilakukan pengujian beberapa variasi jumlah klaster (k). Nilai silhouette score digunakan sebagai dasar pemilihan parameter untuk memperoleh konfigurasi klaster yang paling sesuai. Hasil pengujian untuk setiap nilai k ditampilkan pada Gambar 2.



Gambar 2. Grafik *K-Means* vs. *Silhouette Score*

Berdasarkan Gambar 2, nilai silhouette score cenderung meningkat seiring bertambahnya jumlah klaster dan mencapai nilai tertinggi sebesar 0.14 pada k = 13. Selain memberikan nilai silhouette tertinggi, konfigurasi tersebut juga menghasilkan 13 klaster tanpa klaster kosong dengan jumlah anggota berkisar antara 13 hingga 109 dokumen. Berdasarkan hasil evaluasi, konfigurasi *k* = 13 dipilih sebagai parameter optimal untuk membentuk klaster pada algoritma K-Means.

2) DBSCAN

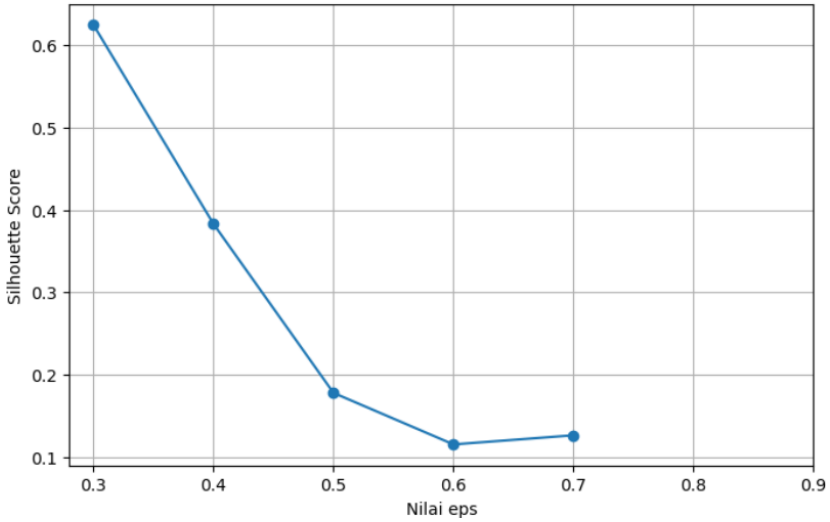
Pada algoritma DBSCAN, dilakukan pengujian beberapa kombinasi nilai ε (eps) dan min_samples untuk memperoleh parameter yang menghasilkan kualitas klaster terbaik. Hasil pengujian menunjukkan bahwa semakin besar nilai ε, semakin banyak dokumen yang tergabung ke dalam klaster sehingga jumlah *noise* berkurang, namun jumlah klaster yang terbentuk cenderung lebih sedikit. Sebaliknya, nilai ε yang lebih kecil menghasilkan lebih banyak klaster dengan jumlah *noise* yang lebih tinggi. Berdasarkan hasil eksperimen, kombinasi parameter ε = 0.6 dan min_samples = 7 menghasilkan struktur klaster yang lebih stabil dengan jumlah *noise* yang relatif rendah. Hasil eksperimen DBSCAN ditampilkan pada Tabel 5.

Tabel 5. Pemilihan Parameter DBSCAN

eps	Klaster	Noise
0.3	3	624
0.4	9	523
0.5	6	299
0.6	5	55
0.7	2	0
0.8	1	0

Berdasarkan Tabel 5, perubahan nilai ε memberikan pengaruh yang signifikan terhadap jumlah klaster dan data *noise* yang dihasilkan. Pada nilai ε yang kecil, sebagian besar dokumen dikategorikan sebagai *noise* sehingga hanya sedikit data yang berhasil dikelompokkan ke dalam klaster. Seiring meningkatnya nilai ε, jumlah *noise* cenderung berkurang karena semakin banyak dokumen yang memenuhi kriteria kepadatan untuk bergabung ke dalam klaster. Namun, peningkatan nilai ε juga menyebabkan jumlah klaster yang terbentuk menjadi lebih sedikit. Hasil ini menunjukkan bahwa pemilihan nilai ε berperan penting dalam menentukan struktur klaster yang dihasilkan oleh DBSCAN.

Setelah diperoleh pola perubahan jumlah klaster dan *noise*, analisis dilanjutkan dengan melihat nilai silhouette score pada setiap nilai ε. Grafik ini digunakan untuk mengetahui kualitas pemisahan klaster yang terbentuk pada masing-masing parameter DBSCAN.



Gambar 3. Grafik DBSCAN vs Silhouette Score

Berdasarkan Gambar 3, nilai silhouette score tertinggi diperoleh pada ε = 0.3 sebesar 0.6248. Namun, konfigurasi tersebut menghasilkan 624 data *noise* dari total 656 dokumen sehingga hanya sebagian kecil data yang berhasil dikelompokkan. Seiring meningkatnya nilai ε, jumlah *noise* berkurang meskipun nilai silhouette score cenderung menurun. Pada penelitian ini, parameter ε = 0.6 dan min_samples = 7 dipilih karena mampu membentuk 5 klaster dengan hanya 55 data *noise*, sehingga menghasilkan pengelompokan yang lebih representatif terhadap keseluruhan dataset. Dengan demikian, pemilihan parameter DBSCAN tidak hanya mempertimbangkan nilai silhouette score, tetapi juga proporsi data yang berhasil dikelompokkan dan jumlah *noise* yang dihasilkan.

3.4 *Evaluation*

Hasil klasterisasi dianalisis menggunakan silhouette score dan *purity* sebagai metrik evaluasi. Selanjutnya, dilakukan interpretasi klaster berdasarkan *top terms* yang terbentuk dari klaster K-Means. Hasil interpretasi tersebut ditampilkan pada Tabel 6.

Tabel 6. Interpretasi Bidang Klaster K-Means

K	Jumlah Topik	Top Terms Utama	Interpretasi Bidang
0	74	['implementasi','network', 'jaring', 'monitoring','grafana', 'mikrotik', 'management','sistem', 'aman', 'server', 'security']	Jaringan & Keamanan
1	109	['aplikasi', 'web', 'game', 'rancang bangun', 'metode', 'sistem', prototype]	Pengembangan Aplikasi & Game
2	87	['aplikasi', 'rancang', 'framework', 'laravel', 'bangun aplikasi sistem', 'codeigniter', 'android']	Pengembangan Aplikasi dengan *Framework*

K	Jumlah Topik	Top Terms Utama	Interpretasi Bidang
3	72	['sistem informasi', 'rancang bangun', 'rancang', 'laravel', 'framework', 'framework laravel', 'web framework']	(Laravel, CodeIgniter) Pengembangan Sistem Informasi dengan Laravel
4	13	['rest', 'api', 'bangun rest api', 'js', 'express js', 'golang']	Pengembangan Backend/API
5	60	['design thinking', 'ux', 'ui', 'rancang ui ux', 'aplikasi']	UI/UX dengan Metode Design Thinking
6	30	['jaring', 'simulasi', 'aman', 'metode', 'otomasi', 'kubernetes', 'pusat', 'tools', 'integrasi', 'perangkat', 'strategi', 'defined', 'software defined']	Jaringan & Otomasi
7	17	['internet of things', 'kontrol', 'sistem', 'sistem monitoring', 'teknologi']	Internet of Things (IoT)
8	26	['rancang sistem informasi', 'sistem informasi', 'analisis', 'web', 'informasi akademik', 'yii', 'service']	Sistem Informasi dengan Yii Framework
9	24	['aplikasi', 'rancang', 'mobile', 'automation', 'testing', 'mobile android']	Pengembangan Aplikasi Mobile & Pengujian
10	75	['kembang', 'kembang aplikasi', 'sistem', 'web', 'js', 'react', 'backend', 'website']	Pengembangan App/Web dengan React Js
11	55	['analisis', 'algoritma', 'data', 'sentimen', 'prediksi', 'visualisasi', 'platform', 'banding', 'machine', 'learning', 'bayes', 'analisis banding', 'metode']	Data Science & Machine Learning
12	14	['centered design', 'user centered design', 'user', 'ux', 'ui', 'website']	UI/UX dengan Metode User Centered Design

Berdasarkan Tabel 6, bidang *Software Engineering* terbagi ke dalam tujuh klaster yang memiliki karakteristik topik berbeda. Kondisi ini menunjukkan bahwa meskipun berada pada bidang yang sama, setiap klaster memiliki istilah dominan yang berbeda, seperti aplikasi, web, API, dan framework. Hasil tersebut sejalan dengan karakteristik TF-IDF yang memberikan bobot lebih tinggi pada term yang bersifat spesifik. Selain itu, dominasi bidang *Software Engineering* juga mencerminkan kondisi aktual di STT-NF, di mana sebagian besar topik tugas akhir berfokus pada pengembangan perangkat lunak.

Selanjutnya, 13 klaster yang dihasilkan K-Means dikelompokkan ke dalam empat bidang peminatan, yaitu *Software Engineering*, *Network Engineering & Cyber Security, UI/UX Design,* dan *Data Engineering*. Jumlah klaster yang lebih besar menunjukkan bahwa satu bidang penelitian dapat terdiri atas beberapa subtopik dengan karakteristik yang berbeda. Oleh karena itu, beberapa klaster yang memiliki *top terms* serupa dikelompokkan ke dalam bidang penelitian yang sama. Sementara itu, hasil klasterisasi DBSCAN menunjukkan bahwa sebagian besar dokumen tergabung pada satu klaster utama yang merepresentasikan bidang *Software Engineering*. Beberapa klaster lain berhasil mengidentifikasi topik yang lebih spesifik, seperti *Extreme Programming, User Centered Design, Internet of Things*, dan monitoring sistem berbasis server. Namun, DBSCAN belum mampu membentuk klaster khusus untuk bidang Data Engineering sebagaimana dihasilkan algoritma K-Means.

Selanjutnya, evaluasi dijalankan menggunakan metrik *purity* untuk mengukur tingkat kesesuaian hasil klasterisasi dengan label asli bidang penelitian (*ground truth*). Untuk keperluan evaluasi eksternal, setiap topik tugas akhir diberi label bidang penelitian secara manual berdasarkan peminatan di Prodi Teknik Informatika STT-NF sebelum proses klasterisasi dilakukan. Perhitungan *purity* mengacu pada Persamaan (6), sedangkan perbandingan hasil klasterisasi K-Means dengan label asli ditampilkan pada Tabel 7.

Tabel 7. Perbandingan Label Topik

label_asli K	DE	NE & CS	SE	UI/UX	Purity
0	4	50	20	0	0.676
1	8	8	79	14	0.725
2	0	0	87	0	1.000
3	0	1	71	0	0.986
4	0	0	13	0	1.000
5	0	0	4	56	0.933
6	4	17	7	2	0.567
7	0	5	12	0	0.706
8	0	0	25	1	0.962
9	0	0	24	0	1.000
10	1	1	72	1	0.960
11	49	1	5	0	0.891
12	0	0	1	13	0.929

Secara keseluruhan, hasil perhitungan mendapatkan nilai *purity* 0.8659, yang menunjukkan bahwa sebagian besar dokumen telah dikelompokkan ke dalam klaster yang sesuai dengan label bidang penelitian. Sementara itu, algoritma DBSCAN memperoleh nilai *purity* sebesar 0.6589, yang mengindikasikan bahwa tingkat kesesuaian hasil klasterisasi terhadap label sebenarnya masih lebih rendah dibandingkan K-Means. Nilai *purity* DBSCAN sebagian dipengaruhi oleh dominasi dokumen bidang *software engineering* yang terkonsentrasi pada satu klaster utama. Ditinjau berdasarkan nilai silhouette score dan *purity*, algoritma K-Means menunjukkan performa lebih baik dibandingkan DBSCAN dalam mengelompokkan topik tugas akhir. Ringkasan hasil evaluasi kedua algoritma disajikan pada Tabel 8.

Tabel 8. Perbandingan Kinerja Algoritma K-Means dan DBSCAN

Metrik Evaluasi	K-Means	DBSCAN (*eps*=0.6, *min_samples*=7)
Jumlah Klaster	13	5
Data *Noise*	0	55
Silhouette Score	0.1434	0.1153
Purity	0.8659	0.6589
Bidang Utama Teridentifikasi	4 Bidang	3 Bidang

Berdasarkan hasil evaluasi, algoritma K-Means memperoleh nilai silhouette score 0.1434 dan *purity* 0.8659, sedangkan DBSCAN memperoleh nilai silhouette score 0.1153 dan *purity* 0.6589. Nilai silhouette score pada kedua algoritma yang berada di rentang positif rendah mengindikasikan bahwa batas antar-klaster masih cenderung saling tumpang tindih (*overlapping*), yang merupakan karakteristik data teks pendek dengan tingkat *sparsity* tinggi. Meskipun demikian, nilai *purity* menunjukkan perbedaan performa yang cukup signifikan. K-Means mencapai *purity* sebesar 86.59%, yang menunjukkan bahwa sebagian besar dokumen berhasil dikelompokkan sesuai dengan label bidang penelitian (*ground truth*). Nilai tersebut menjelaskan bahwa hasil pengelompokan yang dihasilkan tetap memiliki tingkat kesesuaian yang tinggi terhadap label sebenarnya meskipun pemisahan klaster belum sepenuhnya tegas. Sebaliknya, DBSCAN hanya memperoleh *purity* sebesar 65.89%, yang mencerminkan tingkat kesesuaian terhadap label lebih rendah. Berdasarkan kualitas klaster dan tingkat kesesuaiannya terhadap *ground truth*, K-Means menunjukkan performa lebih baik dibandingkan DBSCAN dalam mengelompokkan topik tugas akhir.

3.5 Pembahasan

Meskipun nilai silhouette score kedua algoritma tergolong rendah dibandingkan rentang ideal yang mendekati 1, kondisi ini umum ditemukan pada klasterisasi teks pendek (*short text clustering*). Teks pendek memiliki keterbatasan informasi semantik (*lack of information*) dan tingkat sparsitas (*sparsity*) yang tinggi, sehingga representasi vektor seperti TF-IDF cenderung menghasilkan fitur yang saling tumpang tindih dan menyebabkan batas antar klaster menjadi

kurang tegas [12], [20]. Selain itu, tingginya variasi topik tugas akhir dan kemiripan kosakata antar bidang penelitian turut mempersulit proses pemisahan klaster. Temuan ini sejalan dengan penelitian [21] yang memperoleh nilai silhouette score sebesar 0.12 pada klasterisasi dokumen skripsi dan penelitian [22] yang hanya mencapai 0.05 pada kumpulan paper ilmu komputer.

Meskipun demikian, hasil evaluasi menunjukkan bahwa algoritma K-Means memperoleh nilai silhouette score dan *purity* lebih tinggi dari DBSCAN. Hasil tersebut menunjukkan bahwa tujuan penelitian untuk mengidentifikasi algoritma yang paling sesuai dalam mengelompokkan topik tugas akhir telah tercapai. Kemampuan metrik *purity* dalam merepresentasikan ketepatan pemetaan ini sejalan dengan konsep evaluasi eksternal klaster, yaitu semakin tinggi nilai *purity*, semakin besar proporsi dokumen yang sesuai dengan kelompok dominan dalam satu klaster [10].

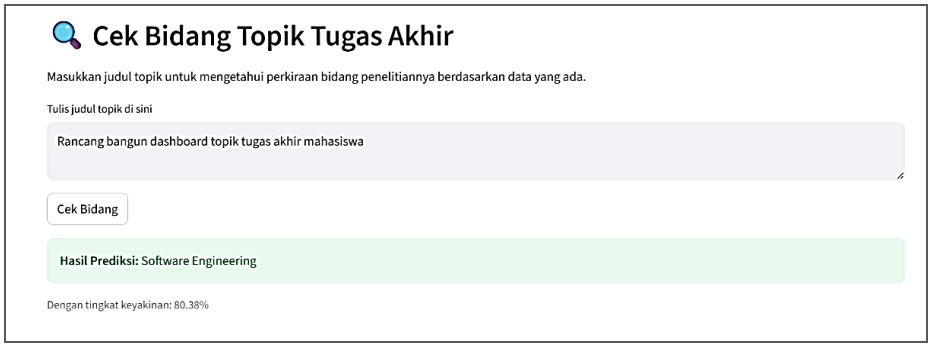
Keunggulan K-Means pada penelitian ini dipengaruhi oleh karakteristik data yang digunakan. Representasi TF-IDF menghasilkan fitur berdimensi tinggi dan relatif tersebar, sehingga pendekatan berbasis *centroid* lebih mampu menangkap pola kemiripan antar-dokumen melalui perhitungan jarak antar data [8]. Sebaliknya, DBSCAN bergantung pada kepadatan data sehingga hasil pengelompokannya sangat dipengaruhi oleh pola persebaran dokumen dan parameter yang digunakan [7]. Akibatnya, sejumlah dokumen teridentifikasi sebagai *noise* sehingga tidak tergabung secara optimal ke dalam klaster utama. Temuan ini mendukung penelitian [5] dan [6] yang menunjukkan bahwa K-Means memiliki performa yang baik pada klasterisasi dokumen akademik, namun berbeda dengan penelitian [7] yang memperoleh hasil optimal menggunakan DBSCAN. Perbedaan tersebut kemungkinan disebabkan oleh perbedaan karakteristik dataset, jumlah dokumen, distribusi data, serta proporsi *noise* yang berbeda pada masing-masing penelitian.

Dari sisi kontribusi ilmiah, penelitian ini memberikan bukti empiris mengenai performa dua pendekatan klasterisasi yang berbeda, yaitu berbasis *centroid* dan densitas. Hasil penelitian menunjukkan K-Means lebih sesuai digunakan untuk proses pemetaan bidang penelitian pada dataset topik tugas akhir mahasiswa yang memiliki karakteristik serupa. Selain itu, implementasi hasil klasterisasi dalam bentuk *dashboard* interaktif memberikan kontribusi praktis dalam mendukung pengelolaan topik tugas akhir dan distribusi dosen pembimbing secara objektif.

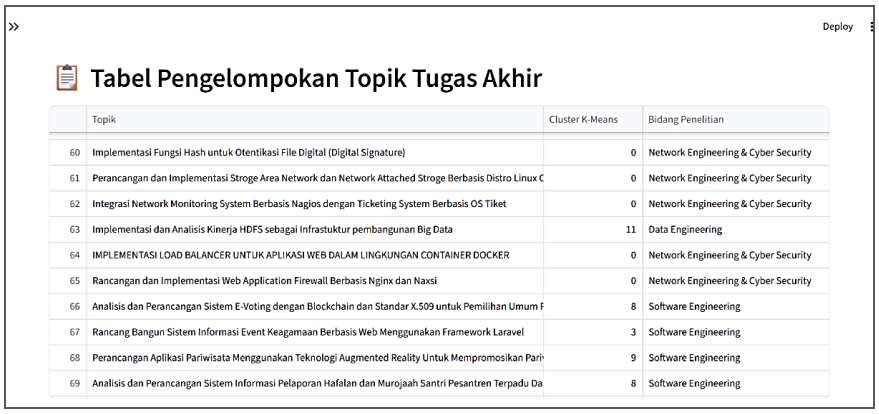
Penelitian ini masih memiliki sejumlah keterbatasan. Data yang digunakan masih terbatas pada satu program studi dan menggunakan representasi TF-IDF, sehingga hubungan makna antar kata belum tergambarkan secara optimal. Untuk penelitian selanjutnya disarankan mengeksplorasi penggunaan pendekatan lain seperti *word embedding, Word2Vec,* maupun *BERT*, serta melakukan pengujian pada dataset yang lebih luas dan bervariasi guna memperoleh hasil yang lebih komprehensif.

3.6 *Deployment*

Tahap deployment diwujudkan melalui *dashboard* interaktif berbasis Streamlit yang menampilkan hasil klasterisasi topik tugas akhir menggunakan algoritma K-Means dan DBSCAN. *Dashboard* menyediakan fitur visualisasi hasil pengelompokan, pengunduhan data klaster, serta prediksi bidang penelitian berdasarkan judul topik tugas akhir yang dimasukkan pengguna. Implementasi ini diharapkan dapat mendukung proses pemetaan bidang penelitian dan penentuan dosen pembimbing. Hasil implementasi dalam bentuk *dashboard* ditampilkan pada Gambar 4 dan 5.



Gambar 4. Fitur Prediksi Topik Tugas Akhir



	Topik	Cluster K-Means	Bidang Penelitian
60	Implementasi Fungsi Hash untuk Otentikasi File Digital (Digital Signature)	0	Network Engineering & Cyber Security
61	Perancangan dan Implementasi Stroge Area Network dan Network Attached Stroge Berbasis Distro Linux C	0	Network Engineering & Cyber Security
62	Integrasi Network Monitoring System Berbasis Nagios dengan Ticketing System Berbasis OS Tiket	0	Network Engineering & Cyber Security
63	Implementasi dan Analisis Kinerja HDFS sebagai Infrastuktur pembangunan Big Data	11	Data Engineering
64	IMPLEMENTASI LOAD BALANCER UNTUK APLIKASI WEB DALAM LINGKUNGAN CONTAINER DOCKER	0	Network Engineering & Cyber Security
65	Rancangan dan Implementasi Web Application Firewall Berbasis Nginx dan Naxsi	0	Network Engineering & Cyber Security
66	Analisis dan Perancangan Sistem E-Voting dengan Blockchain dan Standar X.509 untuk Pemilihan Umum F	8	Software Engineering
67	Rancang Bangun Sistem Informasi Event Keagamaan Berbasis Web Menggunakan Framework Laravel	3	Software Engineering
68	Perancangan Aplikasi Pariwisata Menggunakan Teknologi Augmented Reality Untuk Mempromosikan Pari	9	Software Engineering
69	Analisis dan Perancangan Sistem Informasi Pelaporan Hafalan dan Murojaah Santri Pesantren Terpadu Da	8	Software Engineering

Gambar 5. Tabel Pengelompokan Topik Tugas Akhir

4. Simpulan

Penelitian ini menyimpulkan bahwa algoritma K-Means (k = 13) memiliki performa lebih baik dibandingkan DBSCAN dalam mengelompokkan topik tugas akhir mahasiswa Teknik Informatika STT-NF. K-Means memperoleh nilai silhouette score 0.1434 dan *purity* 0.86585, serta mampu memetakan dokumen ke dalam empat bidang penelitian utama secara representatif. Hasil klasterisasi kemudian diimplementasikan dalam *dashboard* interaktif berbasis Streamlit untuk mendukung pemetaan bidang penelitian dan distribusi dosen pembimbing. Meskipun demikian, hasil klasterisasi masih dipengaruhi oleh karakteristik teks pendek dan keterbatasan representasi TF-IDF dalam menangkap hubungan semantik antar topik. Oleh karena itu, penelitian selanjutnya dapat mengeksplorasi model bahasa seperti BERT atau Word2Vec, metode reduksi dimensi seperti UMAP atau t-SNE, serta melibatkan penilaian pakar dalam proses evaluasi.

Daftar Referensi

[1] N. Widya Utami, I. Gede, and J. E. Putra, “Text Mining Clustering untuk Pengelompokan Topik Dokumen Penelitian Menggunakan Algoritma K-Means dengan Cosine Similarity,” *JINTEKS*, vol. 4, no. 3, pp. 255–259, Aug. 2022.

[2] A. Afda, “Analisa Dokumen Menggunakan Metode TF-IDF,” *J. Ekon. Manaj. Sist. Inf.*, vol. 5, no. 5, pp. 466–470, Mar. 2024, doi: 10.31933/jemsi.v5i5.

[3] S. Rahmah, “Klasterisasi Pola Penjualan Pestisida Menggunakan Metode K-Means Clustering (Studi Kasus di Toko Juanda Tani Kecamatan Hutabayu Raja),” *Djtechno J. Teknol. Inf.*, vol. 1, no. 1, pp. 1–5, 2020.

[4] A. Kristianto, “Analisa Performa K-Means dan DBSCAN dalam Clustering Minat Penggunaan Transportasi Umum,” *J. Ilm. Elektron. dan Komput.*, vol. 14, no. 2, pp. 368–372, Dec. 2021, [Online]. Available: http://journal.stekom.ac.id/index.php/elkom□page368

[5] M. Sholehhudin, M. Fauzi Ali, and S. Adinugroho, “Implementasi Metode Text Mining dan K-Means Clustering untuk Pengelompokan Dokumen Skripsi (Studi Kasus : Universitas Brawijaya),” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 11, pp. 5518–5524, 2018.

[6] D. A. Navastara, E. Mursidah, Y. A. Gonti, D. Wahyuni, P. D. S. Wiyadi, and W. Suadi, “Clustering Topik Penelitian Berbasis Unsupervised Learning untuk Rekomendasi Koleksi Pustaka di Perpustakaan ITS,” *J. Ilm. Teknol. Inf.*, vol. 17, no. 2, pp. 125–134, Aug. 2019.

[7] Z. Vladimir, D. Alamsyah, and W. Widhiarso, “Klasterisasi Topik Skripsi Informatika dengan Metode DBSCAN,” *J. Algoritm.*, vol. 3, no. 1, pp. 82–90, 2022, doi: 10.35957/algoritme.v3i1.3337.

[8] M. R. Muttaqin and M. Defriani, “Algoritma K-Means untuk Pengelompokan Topik Skripsi Mahasiswa,” *Ilk. J. Ilm.*, vol. 12, no. 2, pp. 121–129, 2020, doi: 10.33096/ilkom.v12i2.542.121-129.

[9] R. W. Astuti, B. Badieah, and B. S. WP, “Rancang Bangun Sistem Klasterisasi Dokumen Menggunakan Metode K-Means Untuk Identifikasi Topik Dokumen Tugas Akhir …,” *Pros. Konf. Ilm. Mhs. UNISSULA 2*, vol. 2, pp. 197–205, 2020, [Online]. Available:

http://jurnal.unissula.ac.id/index.php/kimueng/article/view/8588
[10] Amalia, M. S. Lydia, S. D. Fadilla, and M. Huda, "Perbandingan Metode Klaster dan Preprocessing Untuk Dokumen Berbahasa Indonesia," *J. Rekayasa Elektr.*, vol. 14, no. 1, pp. 35–42, Apr. 2018, doi: 10.17529/jre.v14i1.9027.
[11] T. Wurijanto, H. B. Setiawan, and A. B. Tjandrarini, "Penerapan Model CRISP-DM pada Prediksi Nasabah Kredit yang Berisiko Menggunakan Algoritma Support Vector Machine," *J. Ilm. Scroll Jendela Teknol. Inf.*, vol. 10, no. 1, pp. 1–6, 2022, [Online]. Available: https://univ45sby.ac.id/ejournal/index.php/informatika
[12] M. H. Ahmed, S. Tiun, N. Omar, and N. S. Sani, "Short Text Clustering Algorithms, Application and Challenges: A Survey," *Appl. Sci.*, vol. 13, no. 1, p. 342, Jan. 2023, doi: 10.3390/app13010342.
[13] Andre Pratama, J. Nababan, and N. S. Salsabila, "Algoritma Smith-Waterman Untuk Mengidentifikasi Kemiripan Judul Proyek Mahasiswa," *JIKTEKS J. Ilmu Komput. dan Teknol. Inf.*, vol. 3, no. 01, pp. 22–34, Dec. 2024, doi: 10.70404/jikteks.v3i01.125.
[14] Erfian Junianto, Mila Puspitasari, Salman Ilyas Zakaria, Toni Arifin, and Ignatius Wiseto Prasetyo Agung, "Klasifikasi Emosi pada Teks Berbahasa Inggris Menggunakan Pendekatan Ensemble Bagging," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 13, no. 4, pp. 272–281, Nov. 2024, doi: 10.22146/jnteti.v13i4.14440.
[15] A. M. Priyatno and L. Ningsih, "Pembobotan TF-IDF untuk Mendeteksi Akun Spammer di Twitter berdasarkan Tweet dan Representasi Retweet dari Tweet," *Sist. J. Sist. Inf.*, vol. 11, no. 3, pp. 614–622, Sep. 2022, [Online]. Available: http://sistemasi.ftik.unisi.ac.id
[16] M. Rifqi Mubarok, A. Tri, J. Harjanto, and R. Renaldy, "Principal Component Analysis (PCA) dalam Klasterisasi Tingkat Kemiskinan di Indonesia," *J. Inform. Teknol. dan Sains*, vol. 7, no. 3, pp. 1176–1184, Aug. 2025.
[17] R. Adha, N. Nurhaliza, and U. Soleha, "Perbandingan Algoritma DBSCAN dan K-Means Clustering untuk Pengelompokan Kasus Covid-19 di Dunia," *J. Sains, Teknol. dan Ind.*, vol. 18, no. 2, pp. 206–211, Jun. 2021, [Online]. Available: https://covid19.who.int.
[18] R. R. A. Rahman and A. W. Wijayanto, "Pengelompokan Data Gempa Bumi Menggunakan Algoritma DBSCAN," *J. Meteorol. dan Geofis.*, vol. 22, no. 1, p. 31, Oct. 2021, doi: 10.31172/jmg.v22i1.738.
[19] D. Hediyati and I. M. Suartana, "Penerapan Principal Component Analysis (PCA) Untuk Reduksi Dimensi Pada Proses Clustering Data Produksi Pertanian Di Kabupaten Bojonegoro," *J. Inf. Eng. Educ. Technol.*, vol. 5, no. 2, pp. 49–54, 2021, doi: 10.26740/jieet.v5n2.p49-54.
[20] Q. Xu, H. B. Zan, and S. W. Ji, "A lightweight mixup-based short texts clustering for contrastive learning," *Front. Comput. Neurosci.*, vol. 17, no. 11, p. 1334748, 2023, doi: 10.3389/fncom.2023.1334748.
[21] D. Adhe, C. Rachman, R. Goejantoro, F. Deny, and T. Amijaya, "Implementasi Text Mining Pengelompokkan Dokumen Skripsi Menggunakan Metode K-Means Clustering Implementation Of Text Mining For Grouping Thesis Documents Using K-Means Clustering," *J. EKSPONENSIAL*, vol. 11, no. 2, p. 167, 2020.
[22] J. Putra Laksana, H. Irsyad, and A. Rahman, "Analisis Topik Dominan Dalam Paper Ilmu Komputer Menggunakan TF-IDF Dan K-Means," *Bul. Ilm. Inform. Teknol.*, vol. 3, no. 3, pp. 78–84, 2025, doi: 10.58369/biit.v2i3.122.