

Pemodelan Klasifikasi Tingkat Kerusakan Peralatan Penunjang Akademik Berbasis *Natural Language Processing* dan *Naïve Bayes*

¹Trizaurah Armiani, ²Diah Novita Sari, ³Aqila Putri Fathiyah

⁴Rahmawati Dwi Lestari, ⁵Muhammad Karison

^{1*,2,3,4}Manajemen Informatika, Politeknik Negeri Sriwijaya, Palembang, Indonesia

⁵Unit Penunjang Akademik Perawatan dan Pemeliharaan
Politeknik Negeri Sriwijaya, Palembang, Indonesia

*Korespondensi: trizaurah.armiani@polsri.ac.id

Submit : 15 Sept 2026 | Diterima : 12 Okt 2026 | Terbit : 14 Okt 2026

ABSTRACT

The management of academic equipment damage reports faces challenges because damage information is generally recorded as unstructured text with varied descriptions. This condition makes it difficult to identify damage severity consistently and may delay asset handling. This study aims to apply Natural Language Processing to process damage report descriptions and use the Naïve Bayes algorithm to classify equipment damage severity. The dataset was obtained from damage reports at the Unit Penunjang Akademik Perawatan dan Perbaikan of Politeknik Negeri Sriwijaya for the period 2021–2026. A total of 4,147 reports were collected, and 2,374 records remained after data preparation for the classification process. Text processing consisted of case folding, tokenizing, filtering, and stemming, followed by Term Frequency-Inverse Document Frequency (TF-IDF) weighting. Due to the imbalanced class distribution, Synthetic Minority Over-sampling Technique (SMOTE) was applied to the training data before classification using Naïve Bayes. Evaluation on 475 testing records achieved an accuracy of 86.53%. The results indicate that the combination of text preprocessing, TF-IDF, SMOTE, and Naïve Bayes can be applied to classify equipment damage severity based on textual damage reports.

Keywords: Academic Asset, Classification, Naïve Bayes, Natural Language Processing TF-IDF

ABSTRAK

Pengelolaan laporan kerusakan peralatan akademik menghadapi permasalahan karena informasi kerusakan umumnya disampaikan dalam bentuk teks tidak terstruktur dengan variasi deskripsi yang berbeda. Kondisi tersebut menyulitkan proses identifikasi tingkat kerusakan secara konsisten dan dapat memperlambat penanganan aset. Penelitian ini bertujuan menerapkan *Natural Language Processing* untuk mengolah deskripsi laporan kerusakan dan menggunakan algoritma *Naïve Bayes* untuk mengklasifikasikan tingkat kerusakan peralatan. Data penelitian diperoleh dari laporan kerusakan pada Unit Penunjang Akademik Perawatan dan Perbaikan Politeknik Negeri Sriwijaya periode 2021–2026. Sebanyak 4.147 data laporan dikumpulkan dan melalui tahap data preparation diperoleh 2.374 data yang digunakan dalam proses klasifikasi. Pengolahan teks dilakukan melalui case folding, tokenizing, filtering, dan stemming, kemudian data direpresentasikan menggunakan pembobotan Term Frequency-Inverse Document Frequency (TF-IDF). Karena distribusi kelas tidak seimbang, Synthetic Minority Over-sampling Technique (SMOTE) diterapkan pada data training sebelum proses klasifikasi menggunakan *Naïve Bayes*. Evaluasi terhadap 475 data testing menghasilkan akurasi sebesar 86,53%. Hasil tersebut menunjukkan bahwa kombinasi text preprocessing, TF-IDF, SMOTE, dan *Naïve Bayes* dapat diterapkan untuk mengklasifikasikan tingkat kerusakan peralatan berdasarkan teks laporan kerusakan.

Kata Kunci: Aset Akademik, Klasifikasi, *Naïve Bayes*, *Natural Language Processing*, TF-IDF

PENDAHULUAN

Peralatan mesin penunjang akademik berperan dalam menunjang aktivitas akademik dalam lingkungan perguruan tinggi seperti kegiatan perkuliahan, praktikum serta aktivitas lainnya. Kondisi peralatan yang terpelihara dan terjaga dengan baik secara linier memiliki pengaruh positif terhadap kelancaran dalam aktivitas akademik di Politeknik Negeri Sriwijaya. Namun dalam pengelolaannya, proses pelaporan dan pendokumentasian kondisi peralatan yang rusak masih belum secara sepenuhnya melibatkan informasi yang ada di dalam sistem. Kekurangan ini dapat dilihat dari segi aktor yang menggunakan sistem atau dari sistem yang saat ini sedang digunakan yang belum maksimal dalam proses mendapatkan informasi dari kegiatan pelaporan kondisi peralatan yang rusak (Armiani et al., 2026).

Salah satu permasalahan dalam pengelolaan laporan kerusakan adalah informasi yang disampaikan oleh pelapor umumnya berbentuk teks bebas sehingga struktur dan bentuk penyampaian dapat berbeda-beda antara satu laporan dengan laporan lainnya. Pada data penelitian, informasi kerusakan direpresentasikan melalui atribut kerusakan yang memuat deskripsi kondisi peralatan. Dengan berbagai variasi deskripsi menyebabkan informasi kerusakan yang beragam sehingga perlu adanya proses pengolahan sebelum menjadi masukan dalam proses klasifikasi. *Natural Language Processing* (NLP) adalah dapat dimanfaatkan untuk menangani informasi yang disampaikan dalam bentuk teks tanpa format. (Rumaisa et al., 2021) menjelaskan NLP dapat diterapkan untuk mengekstraksi informasi dari teks sebelum digunakan sebagai masukan pada tahap klasifikasi.

Pada data laporan kerusakan, tahapan *text preprocessing* dapat membantu menyeragamkan bentuk teks dan meningkatkan konsistensi sebelum data digunakan dalam proses klasifikasi. Adapun tahapan tersebut seperti *tokenizing*, *filtering*, dan pembobotan menggunakan metode *Term Frequency - Inverse Document Frequency* (TF-IDF), (Liu et al., 2018) menjelaskan bahwa metode ini adalah teknik statistik yang digunakan dalam NLP untuk mengukur seberapa penting sebuah kata dalam suatu dokumen terhadap kumpulan dokumen (korpus) agar dapat dikenali oleh sistem. Hasil perhitungan menggunakan metode TF-IDF selanjutnya digunakan sebagai masukan untuk algoritma *Naïve Bayes* dalam melakukan klasifikasi. Menurut (Pranata et al., 2024) memanfaatkan kombinasi TF-IDF dan *Naïve Bayes* untuk mengklasifikasikan teks *quick count* pada media sosial Twitter. Dalam penelitian lainnya oleh (Mustafa & Alfianti, 2025) menerapkan pendekatan *preprocessing* teks, TF-IDF, dan *Naïve Bayes* untuk mengklasifikasikan berita palsu. Aplikasi dibangun dengan framework Streamlit dan integrasi API Google News. Penelitian-penelitian tersebut menunjukkan pemanfaatan kombinasi metode pengolahan teks dan algoritma klasifikasi pada berbagai jenis kasus. Pada penelitian ini berfokus pada klasifikasi tingkat kerusakan peralatan penunjang akademik berdasarkan deskripsi laporan dan penelitian ini mengolah laporan kerusakan aktual yang memiliki variasi deskripsi teks yang tidak terstruktur untuk mengelompokkan kondisi peralatan ke dalam tingkat kerusakan yaitu ringan, sedang dan berat. Hasil penelitian ini adalah pemodelan dan evaluasi klasifikasi tingkat kerusakan peralatan penunjang akademik dengan mengintegrasikan *text preprocessing*, pembobotan TF-IDF, penanganan ketidakseimbangan kelas menggunakan *Synthetic Minority Over-sampling Technique* (SMOTE), dan algoritma *Naïve Bayes* pada data laporan kerusakan di Politeknik Negeri Sriwijaya.

METODE PENELITIAN

Metode penelitian ini adalah pendekatan NLP dalam mengolah informasi nformasi tekstual pada laporan kerusakan dan mengelompokkannya berdasarkan tingkat kerusakan aset. Pendekatan ini digunakan dalam pengembangan sistem manajemen aset sebagai solusi untuk mengatasi permasalahan subjektivitas dalam penentuan status kerusakan, meningkatkan konsistensi hasil klasifikasi, serta mempercepat proses analisis laporan kerusakan. Proses penentuan kategori kerusakan diawali dengan pengolahan data teks, dilanjutkan dengan representasi fitur, kemudian dilakukan klasifikasi. Setiap deskripsi kerusakan yang diberikan pelapor selanjutnya diproses untuk menentukan salah satu dari tiga kategori tingkat kerusakan. Adapun tiga kategori yaitu:

1. Ringan : Kerusakan yang tidak mengganggu fungsi utama aset
2. Sedang: Kerusakan yang mulai memengaruhi kinerja aset
3. Berat : Kerusakan yang sudah tidak dapat di perbaiki lagi

Pengolahan teks diawali dengan beberapa tahapan *preprocessing*, meliputi *case folding*, *tokenizing*, *filtering* dan *stemming* untuk memperoleh representasi teks yang lebih terorganisasi (Turnandes et al., 2024). Teks hasil *preprocessing* kemudian dikonversi menjadi bobot numerik

melalui metode TF-IDF yang digunakan untuk merepresentasikan tingkat relevansi setiap kata terhadap dokumen. Selanjutnya hasil pembobotan tersebut digunakan sebagai masukan bagi algoritma *Naïve Bayes* untuk menentukan kategori kerusakan berdasarkan probabilitas setiap kategori.

Proses ini dilakukan secara sistematis hingga sistem mampu menghasilkan klasifikasi yang konsisten terhadap data yang diberikan. Berikut tahapan-tahapan adalah sebagai berikut:

1. Persiapan Data (*Data Preparation*)

Dataset yang digunakan dalam penelitian ini diperoleh dari laporan kerusakan aset pada Unit Penunjang Akademik (UPA) Perawatan dan Perbaikan Politeknik Negeri Sriwijaya dengan rentang waktu tahun 2021 sampai dengan 2026. Dataset tersebut terdiri dari 4.147 data laporan kerusakan yang digunakan sebagai bahan dalam proses klasifikasi. Data yang digunakan mencakup berbagai atribut, khususnya deskripsi kerusakan sebagai input utama dalam proses pengolahan teks.

2. Preprocessing Data (*Text Preprocessing*)

Tahap preprocessing dilakukan untuk mempersiapkan data teks sebelum memasuki proses pengolahan berikutnya. Pada tahap ini, teks dibersihkan dan dinormalisasi melalui beberapa tahapan, yaitu case folding, tokenizing, filtering, dan stemming.

a. *Case folding*

Case folding merupakan tahapan untuk menyeragamkan penulisan teks dengan mengonversi seluruh karakter huruf menjadi bentuk huruf kecil (*lowercase*) (Alia et al., 2024). Proses ini bertujuan agar penggunaan huruf kapital dan huruf kecil tidak menghasilkan representasi kata yang berbeda dalam proses pengolahan teks. Contoh penerapan case folding pada data laporan kerusakan ditampilkan pada Tabel 1.

Tabel 1 Case Folding	
Data Mentah	Hasil <i>Case folding</i>
Internet tidak menyala / putus	internet tidak menyala / putus
Adaptor switch rusak	adaptor switch rusak
Tidak dingin	tidak dingin
Sering mati....	sering mati....
Lampu led putus / mati 2 buah	lampu led putus / mati 2 buah

b. *Tokenizing*

Tokenizing merupakan tahapan untuk memisahkan teks menjadi unit-unit kata atau token sehingga setiap bagian teks dapat diproses secara lebih terstruktur. Tahap ini dilakukan untuk memudahkan proses pengolahan dan analisis teks pada tahapan berikutnya (Vayadande et al., 2024). Contoh hasil proses tokenizing pada data laporan kerusakan disajikan pada Tabel 2.

Tabel 2 Hasil <i>Tokenizing</i>	
Data	Hasil <i>Tokenizing</i>
internet tidak menyala / putus	[internet, tidak, menyala, putus]
adaptor switch rusak	[adaptor, switch, rusak]
tidak dingin	[tidak, dingin]
sering mati....	[sering, mati....]
lampu led putus / mati 2 buah	[lampu, led, putus, mati, buah]

c. *Filtering*

Tahap *filtering* dilakukan untuk menyaring unsur teks yang kurang relevan dengan menghilangkan kata-kata umum (*stopword*), seperti "dan", "sering", dan "yang". Proses ini juga mencakup penghapusan tanda baca yang tidak diperlukan, seperti penggunaan titik secara berulang ("...."), serta karakter lain yang tidak memberikan informasi penting bagi proses klasifikasi (Petukhova & Fachada, 2022). Contoh hasil proses filtering ditampilkan pada Tabel 3.

Tabel 3 Hasil Filtering	
Data	Hasil <i>Filtering</i>
[internet, tidak, menyala, putus]	[internet, tidak, menyala, putus]
[adaptor, switch, rusak]	[adaptor, switch, rusak]
[tidak, dingin]	[tidak, dingin]

Data	Hasil <i>Filtering</i>
[sering, mati...]	[mati]
[lampu, led, putus, mati, buah]	[putus, mati]

d. *Stemming*

Stemming merupakan tahapan untuk mereduksi kata ke bentuk dasarnya dengan menghilangkan unsur imbuhan yang terdapat pada kata (Jabbar et al., 2023). Proses ini dilakukan untuk menyamakan berbagai bentuk kata yang memiliki makna serupa sehingga representasi data menjadi lebih sederhana dan dapat mendukung proses klasifikasi. Contoh hasil penerapan *stemming* ditampilkan pada Tabel 4.

Tabel 4 Hasil *Stemming*

Data	Hasil <i>Stemming</i>
menyala	nyala
tidak	tidak
mati	mati
dingin	dingin
putus	putus
rusak	rusak

Hasil keseluruhan proses preprocessing yang mencakup case folding, tokenizing, filtering, dan stemming ditampilkan pada Tabel 5.

Tabel 5 Hasil *Preprocessing*

No	Dataset mentah	Hasil <i>Preprocessing</i>				
		<i>Case folding</i>	<i>Tokenizing</i>	<i>Filtering</i>	<i>Steming</i>	
1	Internet tidak menyala / putus	internet tidak menyala / putus	[internet, tidak, menyala, putus]	[internet, tidak, menyala, putus]	tidak, nyala, putus	
2	Adaptor switch rusak	adaptor switch rusak	[adaptor, switch, rusak]	[adaptor, switch, rusak]	rusak	
3	Tidak dingin	tidak dingin	[tidak, dingin]	[tidak, dingin]	tidak, dingin	
4	Sering mati....	sering mati....	[sering, mati....]	[mati]	mati	
5	Lampu led putus / mati 2 buah	lampu led putus / mati 2 buah	[lampu, led, putus, mati, buah]	[putus, mati]	putus, mati	

3. Transformasi Data (Pembobotan TF-IDF)

Setelah tahap *preprocessing* selesai, teks yang telah dibersihkan dan melalui proses *stemming* selanjutnya dikonversi ke dalam bentuk numerik menggunakan metode TF-IDF (*Term Frequency–Inverse Document Frequency*). Transformasi ini diperlukan karena data teks tidak dapat diproses secara langsung oleh algoritma klasifikasi, sehingga setiap kata perlu direpresentasikan dalam bentuk nilai bobot yang menunjukkan tingkat kepentingannya dalam kumpulan dokumen (Septiani et al., 2022). Nilai TF-IDF diperoleh dengan mempertimbangkan frekuensi kemunculan suatu kata dalam sebuah laporan (*Term Frequency*) serta tingkat kemunculan kata tersebut pada keseluruhan laporan (*Inverse Document Frequency*). Perhitungan bobot TF-IDF dilakukan menggunakan rumus sebagai berikut:

a. Hitung TF (*Term Frequency*)

TF dihitung dengan membagi jumlah kemunculan kata dasar yang telah melalui proses filtering (penyaringan kata) dengan total kata yang tersisa dalam dokumen tersebut. Pada tahap ini, kata benda spesifik dihapus karena penelitian berfokus pada identifikasi status kerusakan berdasarkan kata penciri kondisi.

$$F(t, d) = \frac{\text{Jumlah kata } t \text{ dalam dokumen } d}{\text{Jumlah kata dalam dokumen } d} \quad (1)$$

Keterangan :

TF(t,d) : Frekuensi t kemunculan kata dalam dokumen

b. Hitung IDF (*Inverse Document Frequency*)

IDF berfungsi untuk mengurangi bobot kata yang terlalu sering muncul di semua dokumen dan menaikkan bobot kata yang unik. Rumus yang digunakan adalah $IDF = \log(N/$

DF) dengan jumlah total dokumen (N).

$$IDF(t) = \log \frac{N}{1 + DF(t)}$$

Keterangan:

IDF(t) (*Inverse Document Frequency*) : Nilai yang menunjukkan seberapa jarang kata t muncul di seluruh dokumen.

c. Hitung TF-IDF

Nilai akhir bobot kata didapatkan dengan mengalikan nilai TF dengan IDF, nilai ini yang akan menjadi input bagi algoritma *Naïve Bayes*.

$$TF - IDF(t, d) = TF(t, d) \times IDF(t)$$

Keterangan:

TF-IDF(t,d): Nilai bobot kata t pada dokumen d, menunjukkan seberapa penting kata tersebut dalam dokumen.

TF(t,d) (*Term Frequency*): Frekuensi kemunculan kata t dalam dokumen d.

IDF(t) (*Inverse Document Frequency*): Nilai yang menunjukkan seberapa jarang kata t muncul di seluruh dokumen.

t (*term*): Kata atau istilah yang dianalisis.

d (*document*): Dokumen atau laporan yang sedang diproses.

d. Proses Klasifikasi (*Naïve Bayes*)

Setelah nilai bobot TF-IDF diperoleh, tahap berikutnya adalah menentukan kategori kerusakan menggunakan algoritma *Naïve Bayes*. Metode ini melakukan proses klasifikasi dengan menghitung probabilitas setiap kategori berdasarkan karakteristik kata yang terdapat pada laporan kerusakan. Kategori dengan nilai probabilitas paling tinggi kemudian ditetapkan sebagai hasil klasifikasi (Yang, 2018).

Rumus dasar yang digunakan dalam proses klasifikasi *Naïve Bayes* adalah sebagai berikut:

$$P(C|X) = \frac{P(X|C) \cdot P(C)}{P(X)}$$

Keterangan:

$P(C|X)$: Probabilitas suatu kelas C setelah diketahui data X

$P(X|C)$: Probabilitas munculnya data X pada kelas C

$P(C)$: Probabilitas awal dari kelas C

$P(X)$: Probabilitas munculnya data X

Karena nilai $P(X)$ dianggap sama (konstan) untuk semua kelas, maka perhitungan difokuskan pada nilai pembilang untuk menentukan skor klasifikasi:

$$V_{MAP} = \arg \max_{C_j \in C} P(C_j) \times \prod_{i=1}^n P(x_i|C_j)$$

Keterangan:

V_{MAP} : Kategori yang terpilih (nilai probabilitas tertinggi).

$P(C_j)$: *Prior Probability* atau peluang awal dari kategori j

$P(x_i|C_j)$: *Likelihood* atau peluang kata ke-i muncul pada kategori j.

Pada penelitian ini, terdapat tiga kategori status kerusakan yang digunakan sebagai kelas dalam proses klasifikasi, yaitu:

- C_1 : Kerusakan Ringan merupakan kondisi kerusakan yang tidak mengganggu fungsi utama alat atau fasilitas. Aset masih dapat digunakan sebagaimana mestinya, namun memerlukan perbaikan ringan. Contoh: lampu redup, gangguan kecil yang tidak menghambat penggunaan.
- C_2 : Kerusakan Sedang merupakan kondisi kerusakan yang mulai mengganggu fungsi alat, namun masih dapat digunakan secara terbatas. Perbaikan perlu segera dilakukan agar tidak berkembang menjadi kerusakan yang lebih parah. Contoh: pendingin tidak dingin optimal, jaringan tidak stabil.
- C_3 : Kerusakan Berat merupakan kondisi kerusakan yang menyebabkan alat atau fasilitas

tidak dapat digunakan sama sekali. Kerusakan ini memerlukan penanganan segera karena berdampak langsung terhadap aktivitas pembelajaran atau operasional. Contoh: perangkat mati total, jaringan terputus.

Proses labeling dilakukan secara manual oleh peneliti dengan mengacu pada tingkat keparahan kerusakan yang tercantum dalam laporan. Penentuan kategori mengacu pada dampak kerusakan terhadap fungsi utama aset. Untuk meningkatkan validitas data, proses pelabelan juga disesuaikan dengan hasil observasi serta konsultasi dengan pihak teknis di Unit Penunjang Akademik (UPA) Perawatan dan Perbaikan. Langkah-langkah klasifikasi pada penelitian ini dijabarkan sebagai berikut:

a. Perhitungan *Prior Probability* $P(C_j)$

Tahap pertama adalah menghitung probabilitas awal kemunculan setiap kategori berdasarkan data latih yang tersedia menggunakan rumus:

$$P(C_j) = \frac{|D_j|}{|D|}$$

Keterangan:

D_j : Jumlah data dalam kategori (class) j.

D : Jumlah total seluruh data dalam dataset.

b. Pembentukan Vocabulary

Vocabulary merupakan kumpulan seluruh kata unik hasil Preprocessing, yaitu:

$$V = \{tidak, nyala, putus, rusak, dingin, mati\}$$

Jumlah vocabulary: $|V| = 6$

c. Perhitungan *Likelihood* dengan Bobot TF-IDF

Likelihood pada penelitian ini tidak menggunakan frekuensi kemunculan kata secara mentah, melainkan menggunakan akumulasi bobot TF-IDF. Untuk menghindari nilai nol yang dapat menggagalkan perkalian probabilitas, digunakan pendekatan *Laplacian Smoothing* (Rizky et al., 2024).

$$(x_i|C_j) = \frac{\sum TF - IDF(x_i, C_j) + 1}{\sum TF - IDF(C_j) + |V|}$$

Keterangan:

$\sum TF - IDF(x_i, C_j)$: Total bobot kata x_i pada kategori C_j .

$\sum TF - IDF(C_j)$: Total seluruh bobot kata dalam kategori C_j

$|V|$ (*vocabulary*) : Jumlah total kata unik dalam kosakata

Selanjutnya mencari nilai probabilitas tertinggi dari 3 kategori sebagai hasil klasifikasi pengkategorian status kerusakan.

HASIL DAN PEMBAHASAN

Hasil pengolahan data dan klasifikasi merupakan tahapan yang dilakukan untuk mengubah data laporan kerusakan aset menjadi informasi yang dapat digunakan dalam proses penentuan kategori kerusakan. Proses penelitian diawali dengan tahap persiapan dan pembersihan data, kemudian dilanjutkan dengan pengolahan teks melalui pendekatan *Natural Language Processing* (NLP). Data hasil *preprocessing* selanjutnya direpresentasikan menggunakan pembobotan TF-IDF sebelum digunakan dalam proses klasifikasi dengan algoritma *Naïve Bayes*. Performa model kemudian dievaluasi menggunakan *Confusion Matrix* untuk mengetahui tingkat keberhasilan klasifikasi. Rangkaian tahapan tersebut dilakukan untuk memastikan laporan kerusakan dapat dikelompokkan secara otomatis ke dalam kategori yang telah ditetapkan.

a) *Data Preparation*

Dataset penelitian ini diperoleh dari laporan kerusakan aset pada UPA-PP dengan jumlah awal sebanyak 4.147 data. Pada tahap ini dilakukan beberapa proses, yaitu normalisasi data, penghapusan data kosong (*missing value*), penghapusan data duplikat (*duplicate data*), serta pelabelan kategori kerusakan. Selain itu, dilakukan seleksi atribut dengan mengambil kolom

Kerusakan/Keluhan sebagai atribut utama yang digunakan dalam proses klasifikasi karena kolom tersebut berisi informasi mengenai kondisi kerusakan aset yang dilaporkan oleh pengguna.

Proses normalisasi dilakukan untuk menyeragamkan format data sehingga lebih mudah diproses pada tahapan berikutnya. Selanjutnya dilakukan pemeriksaan terhadap data yang memiliki nilai kosong (null value) dan data yang terduplikasi. Data yang tidak memenuhi kriteria tersebut dihapus dari dataset agar tidak memengaruhi hasil klasifikasi. Setelah proses pembersihan data selesai, dilakukan pelabelan secara manual terhadap setiap data laporan berdasarkan tingkat kerusakan yang dilaporkan.

Kategori kerusakan yang diterapkan dalam penelitian ini dikelompokkan ke dalam tiga kelas, yaitu:

- C1 (Ringan), yaitu kerusakan yang masih dapat digunakan atau tidak mengganggu fungsi utama aset secara signifikan.
- C2 (Sedang), yaitu kerusakan yang menyebabkan fungsi aset terganggu dan memerlukan perbaikan.
- C3 (Berat), yaitu kerusakan yang menyebabkan aset tidak dapat digunakan atau memerlukan penanganan khusus maupun penggantian.

Setelah seluruh tahapan data preparation dilakukan, jumlah data yang dapat digunakan dalam proses klasifikasi berkurang dari 4.147 data menjadi 2.374 data.

b) *Preprocessing* teks

Preprocessing teks dilakukan sebagai tahap awal untuk mempersiapkan data sebelum memasuki proses klasifikasi. Pada penelitian ini, proses tersebut diterapkan pada kolom Kerusakan/Keluhan yang memuat uraian mengenai kondisi kerusakan aset. Pengolahan teks mencakup beberapa tahapan, yaitu *case folding* untuk menyeragamkan huruf menjadi huruf kecil, *tokenizing* untuk memisahkan teks ke dalam token, *filtering* untuk menyaring *stopword* yang kurang relevan, dan *stemming* untuk memperoleh bentuk dasar dari kata berimbuhan.

Rangkaian *preprocessing* bertujuan menghasilkan teks yang lebih terstruktur, seragam, dan relevan sehingga dapat mendukung proses pembobotan TF-IDF serta klasifikasi menggunakan algoritma *Naïve Bayes*. Hasil keseluruhan tahapan *preprocessing* teks ditampilkan pada Gambar 1.

	text_normalized	text_casefold	tokenizing	stopword	stemming
0	jarintidakn internet_mati internet tidak menya...	jarintidakn internet_mati internet tidak menya...	[jarintidakn, internet_mati, internet, tidak, ...]	[jarintidakn, internet_mati, internet, menya...]	[jarintidakn, internet, mati, internet, nyal, ...]
1	jaringan internet_mati...	jaringan internet_mati...	[jaringan, internet_mati]	[jaringan, internet, mati]	[jaring, internet mati]
2	ac (tidak_dingin/tidak_dingin) tidak_dingin	ac (tidak_dingin/tidak_dingin) tidak_dingin	[ac, tidak_dingin, tidak_dingin, tidak_dingin]	[ac, tidak_dingin, tidak_dingin, tidak_dingin]	[ac, tidak_dingin, tidak_dingin, tidak_dingin]
3	ac (tidak_dingin/tidak_dingin) tidak_dingin	ac (tidak_dingin/tidak_dingin) tidak_dingin	[ac, tidak_dingin, tidak_dingin, tidak_dingin]	[ac, tidak_dingin, tidak_dingin, tidak_dingin]	[ac, tidak_dingin, tidak_dingin, tidak_dingin]
4	ac (pipa freon patah) mati hidup	ac (pipa freon patah) mati hidup	[ac, pipa, freon, patah, mati, hidup]	[ac, pipa, freon, patah, mati, hidup]	[ac, pipa, freon, patah, mati, hidup]
5	ac split - ruang meeting t.komputer sering mati	ac split - ruang meeting t.komputer sering mati	[ac, split, ruang, meeting, t, komputer, sering, mati]	[ac, split, ruang, meeting, t, komputer, sering, ...]	[ac, split, ruang, meeting, t, komputer, sering, ...]
6	ac tidak_dinginonic (split) ruang lab dasar te...	ac tidak_dinginonic (split) ruang lab dasar te...	[ac, tidak_dinginonic, split, ruang, lab, dasa...]	[ac, tidak_dinginonic, split, ruang, lab, dasa...]	[ac, tidak_dinginonic, split, ruang, lab, dasa...]
7	micropon jaringan habis baterai micropon jari...	micropon jaringan habis baterai micropon jari...	[micropon, jaringan, habis, baterai, micropon, ...]	[micropon, jaringan, habis, baterai, micropon, ...]	[micropon, jaring, habis, baterai, micropon, ...]

Gambar 1 Hasil *Preprocessing* Teks

c) Ekstraksi Fitur TF-IDF

Setelah tahap preprocessing selesai, data teks yang telah dinormalisasi selanjutnya dikonversi menjadi representasi numerik melalui metode TF-IDF. Metode ini digunakan untuk memberikan nilai bobot pada setiap kata dengan mempertimbangkan frekuensi kemunculannya dalam suatu dokumen serta tingkat kemunculannya pada keseluruhan dokumen dalam dataset.

Penerapan TF-IDF memungkinkan data teks yang semula tidak terstruktur direpresentasikan dalam bentuk numerik sehingga dapat digunakan sebagai masukan bagi algoritma *Naïve Bayes*. Nilai bobot yang lebih tinggi menunjukkan bahwa suatu kata memiliki kontribusi yang lebih besar dalam merepresentasikan karakteristik dokumen.

Hasil proses transformasi TF-IDF ditampilkan pada Gambar 2. Tahap ini menghasilkan matriks fitur yang merepresentasikan setiap dokumen dalam bentuk nilai numerik. Matriks fitur tersebut selanjutnya digunakan sebagai masukan pada proses klasifikasi menggunakan algoritma *Naïve Bayes* untuk menentukan kategori tingkat kerusakan aset berdasarkan deskripsi yang terdapat pada laporan pengguna.

Clean Text	TF-IDF
jarintidakn internet mati internet nyala putus	internet (0.795), jarintidakn (0.353), mati (0.153), nyala (0.370), putus (0.289)
jaring internet mati	internet (0.644), jaring (0.723), mati (0.248)
ac tidak dingin tidak dingin tidak dingin	ac (0.159), dingin (0.771), tidak (0.617)
ac tidak dingin tidak dingin tidak dingin	ac (0.159), dingin (0.771), tidak (0.617)
ac pipa freon patah mati hidup	ac (0.197), freon (0.436), hidup (0.399), mati (0.180), patah (0.588), pipa (0.484)
ac split ruang meeting t komputer sering mati	ac (0.170), komputer (0.357), mati (0.155), meeting (0.588), ruang (0.347), sering (0.500), split (0.322)
ac tidak dinginonic split ruang lab dasar telekomunikasi lt tidak dingin	ac (0.136), dasar (0.321), dingin (0.219), dinginonic (0.199), lab (0.367), lt (0.355), ruang (0.277), split (0.258), telekomunikasi (0.513), tidak (0.350)
ropon jaring habis baterai micropon jaring mati nyala	baterai (0.282), habis (0.286), jaring (0.510), mati (0.088), micropon (0.725), nyala (0.211)
komputer errorr loading	errorr (0.495), komputer (0.489), loading (0.718)
ssette tidak dinginonic ruang dosen mati ulang hidup nyala	ac (0.163), cassette (0.410), dinginonic (0.239), dosen (0.468), hidup (0.330), mati (0.149), nyala (0.360), ruang (0.333), tidak (0.210), ulang (0.337)

Gambar 2 Hasil Ekstraksi Fitur

d) Metode *Synthetic Minority Over-sampling Technique* (SMOTE)

Hasil pelabelan menunjukkan bahwa jumlah data pada masing-masing kategori kerusakan belum terdistribusi secara merata. Kategori C2 memiliki jumlah data yang lebih dominan dibandingkan dengan C1 dan C3 sehingga dataset termasuk dalam kondisi imbalanced. Ketidakseimbangan tersebut berpotensi membuat model lebih banyak mengenali kelas mayoritas dan kurang optimal dalam mengidentifikasi data dari kelas minoritas.

Untuk menangani ketidakseimbangan kelas tersebut, SMOTE (*Synthetic Minority Over-sampling Technique*) diterapkan pada data training. Metode ini menambah jumlah sampel pada kelas minoritas dengan membentuk data sintesis berdasarkan karakteristik data yang tersedia, sehingga distribusi antar kelas pada data training menjadi lebih seimbang (Dablain et al., 2023).

e) *Naïve Bayes*

Data training yang telah diseimbangkan melalui SMOTE selanjutnya digunakan dalam proses pembentukan model klasifikasi *Naïve Bayes*. Data yang digunakan telah melewati tahapan preprocessing, transformasi TF-IDF, dan penyeimbangan kelas. Model yang dihasilkan kemudian diterapkan pada data testing sebanyak 475 data untuk memperoleh hasil prediksi.

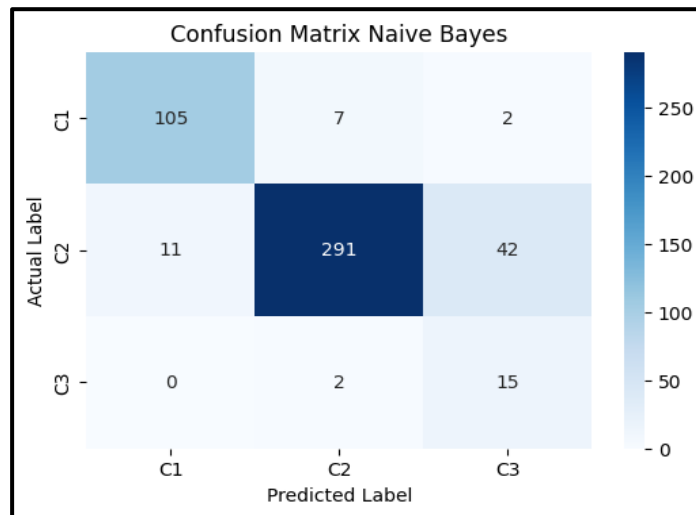
Naïve Bayes merupakan metode klasifikasi yang menggunakan pendekatan probabilitas untuk menentukan kelas suatu data berdasarkan karakteristik fitur yang dimiliki. Dalam penelitian ini, model digunakan untuk mengelompokkan laporan kerusakan aset ke dalam tiga kategori, yaitu C1 (Ringan), C2 (Sedang), dan C3 (Berat). Hasil klasifikasi yang diperoleh melalui algoritma *Naïve Bayes* disajikan pada Gambar 3.

	Laporan	Label Aktual	Label Prediksi
1175	lampu penerantidakn tambah pengtidakntian pcs	C2	C1
1027	stop kontak inbow rusak instalasi listrik stop...	C2	C2
1929	internet mati internet mati	C3	C3
566	baik instalasi listrik stop kontak rusak	C2	C2
188	printer dalam printer dapat telur telur semut	C2	C2
1732	listrik stop kontak rusak	C2	C2
1102	kabel panjang stop kontak kabel putus ditentidakh	C2	C3
1313	lampu penerantidakn led bulb philips fitting l...	C1	C1
1019	mcb rusak mcb rusak	C2	C2
1675	listrik lampu penerantidakn led bulb philips r...	C1	C1

Gambar 3 Hasil Algoritma *Naïve Bayes*

f) *Confusion Matrix*

Confusion Matrix digunakan untuk menilai performa model klasifikasi dengan membandingkan hasil prediksi model terhadap kelas aktual pada data pengujian ((Hidayah, 2024). Hasil evaluasi ini menunjukkan jumlah prediksi yang tepat maupun kesalahan prediksi untuk setiap kategori tingkat kerusakan. Visualisasi *Confusion Matrix* pada penelitian ini disajikan pada Gambar 4.

Gambar 4 *Confusion Matrix*

Berdasarkan hasil *Confusion Matrix* pada Gambar 4, performa prediksi pada masing-masing kategori dapat dijelaskan sebagai berikut. Pada kelas C1, sebanyak 105 data diklasifikasikan dengan tepat sebagai C1, sementara 7 data diprediksi sebagai C2 dan 2 data sebagai C3. Pada kelas C2, model berhasil mengklasifikasikan 291 data secara tepat sebagai C2, sedangkan 11 data diprediksi sebagai C1 dan 42 data sebagai C3. Sementara itu, pada kelas C3 terdapat 15 data yang berhasil diprediksi dengan benar, 2 data diklasifikasikan sebagai C2, dan tidak terdapat data yang diprediksi sebagai C1.

Untuk mengetahui performa model secara lebih komprehensif, evaluasi selanjutnya dilakukan menggunakan beberapa metrik, yaitu *precision*, *recall*, *F1-score*, dan *accuracy*. Hasil perhitungan metrik evaluasi tersebut disajikan pada Gambar 5

	precision	recall	f1-score	support
C1	0.91	0.92	0.91	114
C2	0.97	0.85	0.90	344
C3	0.25	0.88	0.39	17
accuracy			0.87	475
macro avg	0.71	0.88	0.74	475
weighted avg	0.93	0.87	0.89	475

Gambar 5 Metrik Evaluasi

Hasil pengujian Hasil pengujian menunjukkan bahwa model *Naïve Bayes* mencapai akurasi sebesar 86,53%. Pada masing-masing kelas, nilai *precision* yang diperoleh adalah 0,91 untuk C1, 0,97 untuk C2, dan 0,25 untuk C3. Sementara itu, nilai *recall* mencapai 0,92 pada C1, 0,85 pada C2, dan 0,88 pada C3. Adapun nilai *F1-score* berturut-turut sebesar 0,91 untuk C1, 0,90 untuk C2, dan 0,39 untuk C3. Perbedaan nilai evaluasi tersebut menunjukkan bahwa kinerja model belum merata pada seluruh kategori tingkat kerusakan. Meskipun akurasi keseluruhan mencapai 86,53%, nilai tersebut belum sepenuhnya merepresentasikan kemampuan model dalam membedakan setiap kategori, khususnya kerusakan berat. Rendahnya nilai *precision* pada kelas C3 dapat dijelaskan melalui analisis *confusion matrix*. Dari 475 data pengujian, terdapat 59 laporan yang diprediksi sebagai kerusakan berat, tetapi hanya 15 laporan yang sesuai dengan label aktual C3. Sebanyak 42 laporan yang sebenarnya termasuk kategori kerusakan sedang (C2) dan dua laporan kategori kerusakan ringan (C1) salah diklasifikasikan sebagai C3. Dengan demikian, jumlah prediksi positif yang tidak sesuai dengan label aktual pada kelas C3 mencapai 44 laporan, sehingga nilai *precision* hanya sebesar 0,25. Sebaliknya, dari 17 laporan yang secara aktual termasuk kategori C3, sebanyak 15 laporan berhasil diidentifikasi dengan benar, menghasilkan nilai *recall* sebesar 0,88. Hasil ini memperlihatkan bahwa model relatif mampu menemukan laporan kerusakan berat, tetapi belum cukup selektif dalam membedakan laporan kerusakan berat dari laporan kerusakan sedang. Pola kesalahan tersebut mengindikasikan bahwa batas klasifikasi antara kategori kerusakan sedang dan berat masih menjadi tantangan bagi model. Salah satu kemungkinan penjelasannya adalah adanya kemiripan istilah atau deskripsi pada laporan kerusakan yang berbeda tingkat keparahannya.

KESIMPULAN

Hasil evaluasi menunjukkan bahwa model *Naïve Bayes* yang mengintegrasikan *text preprocessing* dan pembobotan TF-IDF mampu mengklasifikasikan laporan kerusakan peralatan penunjang akademik dengan akurasi sebesar 86,53%. Namun, performa model belum merata pada seluruh kategori kerusakan. Kelas C1 (ringan) memperoleh *precision*, *recall*, dan F1-score masing-masing sebesar 0,91, 0,92, dan 0,91, sedangkan kelas C2 (sedang) memperoleh nilai masing-masing sebesar 0,97, 0,85, dan 0,90. Pada kelas C3 (berat), model menghasilkan *recall* sebesar 0,88, tetapi nilai *precision* dan F1-score masing-masing hanya sebesar 0,25 dan 0,39. Hasil tersebut menunjukkan bahwa model mampu mengenali sebagian besar laporan yang secara aktual termasuk kategori kerusakan berat, tetapi masih menghasilkan sejumlah besar prediksi kerusakan berat yang tidak sesuai dengan label aktual. Dengan demikian, model berpotensi digunakan sebagai alat bantu pengelompokan awal laporan kerusakan secara otomatis, tetapi hasil prediksinya, khususnya pada kelas C3, masih memerlukan verifikasi sebelum digunakan untuk mendukung penentuan prioritas pemeliharaan.

REFERENSI

- Alia, P. A., Febriana, R. W., Prayogo, J. S., & Kriswibowo, R. (2024). *Implementation Chatbot on Whatsapp Using Artificial Intelligence With Natural Language Processing Method*. 5(1), 8–14.
- Armiani, T., Sari, D. N., Sukha, H. A., Lestari, R. D., Fathiyah, A. P., Informatika, J. M., Sriwijaya, P. N., Peralatan, M., & Sriwijaya, N. (2026). *Implementasi Sistem Manajemen dan Monitoring Peralatan Mesin Penunjang Akademik Pada Politeknik Negeri Sriwijaya*. 71–80.
- Dablain, D., Krawczyk, B., & Chawla, N. V. (2023). DeepSMOTE: Fusing Deep Learning and SMOTE for Imbalanced Data. *IEEE Transactions on Neural Networks and Learning Systems*, 34(9), 6390–6404. <https://doi.org/10.1109/TNNLS.2021.3136503>
- Hidayah, N. (2024). *Implementasi Algoritma Multinomial Naïve Bayes, TF-IDF dan Confusion Matrix dalam Pengklasifikasian Saran Monitoring dan Evaluasi Mahasiswa Terhadap Dosen Teknik Informatika Universitas Dayanu Ikhsanuddin*. 10, 8–15.
- Jabbar, A., Iqbal, S., Tamimy, M. I., Rehman, A., Member, S., & Bahaj, S. A. L. I. (2023). *An Analytical Analysis of Text Stemming Methodologies in Information Retrieval and Natural Language Processing Systems*. 11(December).
- Liu, C., Sheng, Y., Wei, Z., & Yang, Y. (2018). Research of Text Classification Based on Improved TF-IDF Algorithm. *2018 IEEE International Conference of Intelligent Robotic and Control Engineering (IRCE)*, 2, 218–222.
- Mustafa, H. A. T. F., & Alfianti, H. (2025). Klasifikasi berita palsu berbahasa Indonesia menggunakan algoritma Naive Bayes berbasis web. *J. Sains Inform. Terap.(JSIT)*, 4(3), 657–663.
- Petukhova, A., & Fachada, N. (2022). SoftwareX TextCL: A Python package for NLP preprocessing tasks. *SoftwareX*, 19, 101122. <https://doi.org/10.1016/j.softx.2022.101122>
- Pranata, A., Rudiman, R., & Verdikha, N. A. (2024). Klasifikasi Teks Quick Count Pemilihan Presiden 2024 pada Twitter menggunakan Metode TF-IDF dan Naive Bayes. *Jurnal Informatika Terpadu*, 10(2), 93–100.
- Rizky, Y. A., Aziz, A., & Harianto, W. (2024). *Implementasi Naive Bayes Dengan Menggunakan Metode Laplace Smoothing*. 6(3).
- Rumaisa, F., Puspitarani, Y., Rosita, A., & Zakiah, A. (2021). *Penerapan Natural Language Processing (NLP) Di Bidang Pendidikan*. 01(03), 232–235.
- Septiani, D., Isabela, I., Teknik, F., Jakarta, U. N., Teknik, F., & Jakarta, U. N. (n.d.). *Analisis term frequency inverse document frequency (tf-idf) dalam temu kembali informasi pada dokumen teks*. 25, 81–88.
- Turnandes, Y., Irwanda, A. A., & Afrilli, R. (2024). *Analisis Sentimen pada Ulasan Kegiatan Seminar Nasional Sistem Informasi dan Teknologi Komputer 2023 menggunakan Natural Language Processing (NLP)*. 4(2), 101–109.
- Vayadande, K., Kale, D. R., Nalavade, J., Kumar, R., & Magar, H. D. (2024). Text generation \& classification in nlp: A review. *How Machine Learning Is Innovating Today's World: A Concise Technical Guide*, 25–36.
- Yang, F. (2018). An Implementation of Naive Bayes Classifier. *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, 301–306. <https://doi.org/10.1109/CSCI46756.2018.00065>