

Analisis Perbandingan SVM dan *Logistic Regression* dalam Penentuan Prioritas Tiket Dukungan Pelanggan

Larasati Mya Mulyono¹, Najibah Aisyah Muhaa², Indira Fildah Fakhrrana³, Betha Nurina Sari⁴

^{1, 2, 3}Sistem Informasi, Fakultas Ilmu Komputer, Universitas Singaperbangsa Karawang

⁴Informatika, Fakultas Ilmu Komputer, Universitas Singaperbangsa Karawang

¹larasati.mulyono20@gmail.com, ²[najibahisyahibah@gmail.com](mailto:najibahaisyahibah@gmail.com), ³raraindira79@gmail.com,

⁴betha.nurina@staff.unsika.ac.id

Abstract

Automating customer support ticket prioritization remains challenging due to the high similarity of textual descriptions across urgency levels, which limits the discriminative capability of conventional text classification models. This study evaluates the performance of Support Vector Machine (SVM) and Logistic Regression for ticket urgency classification using the CRISP-DM framework. A secondary Kaggle dataset comprising 8,649 records and 17 attributes was utilized. Ticket descriptions were preprocessed and represented using TF-IDF, and priority levels were transformed into binary classes. The results indicate limited classification performance: Logistic Regression achieved 51.33% accuracy with an F1-score of 0.508, while SVM obtained 48.80% accuracy with an F1-score of 0.488. These findings demonstrate that TF-IDF-based textual features alone are insufficient for effectively distinguishing ticket urgency levels and highlight the need for integrating semantic representations and non-textual metadata to enhance prioritization performance.

Keywords: *text mining, SVM, logistic regression, ticket prioritization, customer support, classification*

Abstrak

Otomatisasi penentuan prioritas tiket dukungan pelanggan menghadapi tantangan karena deskripsi teks antar tingkat urgensi memiliki pola yang mirip sehingga membatasi efektivitas model klasifikasi konvensional. Penelitian ini mengevaluasi kinerja *Support Vector Machine* (SVM) dan *Logistic Regression* dalam klasifikasi urgensi tiket menggunakan kerangka CRISP-DM. Dataset sekunder dari Kaggle yang terdiri dari 8.649 data dan 17 atribut digunakan dalam penelitian ini. Deskripsi tiket diproses melalui preprocessing dan direpresentasikan menggunakan TF-IDF, kemudian tingkat prioritas disederhanakan menjadi klasifikasi biner. Hasil menunjukkan performa yang relatif rendah, di mana Logistic Regression memperoleh akurasi 51,33% dan F1-score 0,508, sedangkan SVM memperoleh akurasi 48,80% dan F1-score 0,488. Temuan ini menunjukkan keterbatasan fitur berbasis TF-IDF dalam memprediksi urgensi tiket serta menegaskan perlunya integrasi representasi semantik dan metadata non-teks.

Kata kunci: *text mining, SVM, logistic regression, prioritas tiket, dukungan pelanggan, klasifikasi*

© 2026 Author
Creative Commons Attribution 4.0 International License



1. Pendahuluan

Perkembangan teknologi informasi mendorong peningkatan interaksi antara pelanggan dan penyedia layanan melalui sistem *customer support ticket* [1].

Volume tiket pada perusahaan IT telah meningkat secara signifikan karena upaya digitalisasi yang saat ini dilakukan di semua industri [2]. Peningkatan volume tiket yang signifikan mengakibatkan produk digital menjadi kompleks sehingga menimbulkan

tantangan dalam mengelola dan menentukan prioritas penanganan. Penentuan prioritas yang masih dilakukan secara manual sering kali memakan waktu, tidak konsisten, dan sulit diterapkan pada skala besar, sehingga memperlambat waktu tanggapan terhadap tiket dengan prioritas tinggi dan menurunkan kepuasan pelanggan [3]. Oleh karena itu, diperlukan pendekatan otomatis berbasis *machine learning* yang mampu mengklasifikasikan prioritas tiket secara lebih konsisten dan akurat.

Text mining merupakan proses menggali pengetahuan baru dengan mengekstraksi informasi secara otomatis dari kumpulan teks berukuran besar yang bersifat tidak terstruktur [4]. Salah satu tugas utama yang banyak digunakan dalam *text mining* adalah klasifikasi teks. Klasifikasi teks merupakan proses menetapkan label atau kategori tertentu pada suatu dokumen berdasarkan isi yang dikandungnya. Proses ini melibatkan pembangunan model pembelajaran mesin yang mampu mengenali pola dalam teks untuk menghasilkan keputusan kategoris [5]. Untuk melakukan hal tersebut, diperlukan algoritma yang efektif dalam mengolah data teks, seperti algoritma *Support Vector Machine* (SVM) dan *Logistic Regression*.

Support Vector Machine (SVM) merupakan algoritma yang beroperasi dengan mencari *hyperplane* paling optimal untuk memisahkan dua kelas dalam ruang input berdasarkan prinsip *Structural Risk Minimization* (SRM). Metode ini dikenal memiliki kestabilan serta kemampuan generalisasi yang lebih baik dibandingkan banyak algoritma klasifikasi lainnya [6]. Keunggulan utama dari algoritma ini adalah kemampuan menangani kasus non-linear secara efisien melalui fungsi kernel [7]. Sedangkan, *Logistic Regression* merupakan jenis analisis regresi yang digunakan untuk memodelkan hubungan antara variabel dependen dan satu atau lebih variabel independen dalam bentuk keluaran kategorikal, seperti 0 dan 1 atau benar dan salah. Metode ini menghubungkan variabel bebas dengan variabel terikat yang bersifat biner, sehingga membedakannya dari regresi linear berganda maupun regresi linear lainnya yang digunakan untuk memprediksi nilai kontinu [8].

Beberapa penelitian terdahulu menunjukkan bahwa kinerja SVM dan *Logistic Regression* pada klasifikasi teks dapat bervariasi bergantung pada karakteristik dataset. Pada analisis sentimen ulasan *Mobile Legends*, SVM memperoleh akurasi tertinggi dan performa paling stabil [9]. Sedangkan penelitian pada ulasan aplikasi Netflix menunjukkan bahwa *Logistic Regression* sedikit lebih unggul dibandingkan SVM [10]. Pada studi klasifikasi tweet bencana, SVM kembali memberikan akurasi tertinggi, sementara *Logistic Regression* menunjukkan peningkatan signifikan setelah penanganan ketidakseimbangan kelas menggunakan SMOTE [11], menandakan bahwa algoritma ini lebih sensitif terhadap distribusi

kelas. Temuan ini sejalan dengan rangkaian penelitian lain yang membandingkan kedua algoritma pada berbagai domain teks. Studi mengenai sentimen Tiktokshop dengan judul “Perbandingan Naive Bayes, Support Vector Machine, Logistic Regression dan Random Forest Dalam Mengalisis Sentimen Mengenai Tiktokshop” menempatkan SVM dengan akurasi tertinggi sebesar 81%. Pada deteksi bot dan penipuan tiket konser, SVM juga mendominasi dengan akurasi 91,27% sementara *Logistic Regression* mencapai 90,03%, serta peningkatan performa terlihat setelah penerapan SMOTE pada kelas yang tidak seimbang [12]. Namun, beberapa penelitian seperti analisis sentimen pascapemilu menemukan bahwa *Logistic Regression* mampu memberikan hasil lebih baik dengan akurasi 84,39%, menegaskan bahwa efektivitas algoritma sangat dipengaruhi oleh struktur data [13]. Pada ulasan aplikasi retail, kedua algoritma sama-sama mencapai akurasi 0,87, tetapi *Logistic Regression* menunjukkan *F1-score* lebih tinggi dan tingkat misklasifikasi lebih rendah [14]. Selain itu, studi NLP pada klasifikasi tiket pelanggan menunjukkan bahwa SVM memberikan kinerja paling konsisten pada data teks dukungan pelanggan [15], dan tinjauan sistematis terhadap 41 penelitian menegaskan bahwa SVM merupakan algoritma yang paling sering digunakan dengan akurasi rata-rata 80–90% dalam pemrosesan tiket dukungan pelanggan. Secara keseluruhan, hasil penelitian terdahulu menunjukkan bahwa baik SVM maupun *Logistic Regression* memiliki kekuatan masing-masing dalam klasifikasi teks, namun performanya sangat dipengaruhi oleh karakteristik data, tingkat ketidakseimbangan kelas, dan teknik pra-pemrosesan yang digunakan, sehingga masih diperlukan analisis lebih mendalam pada konteks penentuan prioritas tiket dukungan pelanggan.

Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk menganalisis dan membandingkan kinerja algoritma *Support Vector Machine* (SVM) dan *Logistic Regression* dalam penentuan prioritas tiket dukungan pelanggan. Dengan memahami efektivitas kedua algoritma, perusahaan dapat mengembangkan sistem otomatisasi yang lebih akurat dan efisien untuk mengelola tiket berdasarkan tingkat urgensinya. Penelitian ini diharapkan mampu memberikan wawasan baru mengenai bagaimana data teks pada tiket dukungan pelanggan dapat dimanfaatkan untuk mendukung proses pengambilan keputusan operasional, sekaligus membantu meningkatkan kualitas layanan melalui penanganan tiket yang lebih cepat dan tepat sasaran.

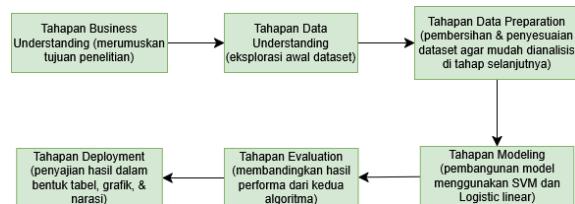
2. Metode Penelitian

Penelitian ini menggunakan metode CRISP-DM (*Cross-Industry Standard Process for Data Mining*) sebagai kerangka kerja utama. Pemilihan CRISP-DM didasarkan pada sifatnya yang terstruktur, iteratif, dan telah teruji secara luas untuk proyek data mining dan

analitik prediktif [16]. Pendekatan siklus hidup CRISP-DM mencakup enam tahapan utama yang sistematis, dimulai dari pemahaman bisnis hingga evaluasi dan penyebaran model [17]. Tahapan-tahapan ini mencakup penentuan tujuan bisnis, evaluasi situasi terkini, pengumpulan data, pembersihan dan transformasi data, penerapan algoritma, hingga penilaian kinerja model yang dihasilkan [18].

Jenis dataset yang dipakai pada penelitian ini merupakan jenis data sekunder yang berasal dari situs Kaggle. Dataset ini terdiri dari 17 kolom data yang mencakup tentang informasi pelanggan (*Customer Name, Customer Email, Customer Age, Customer Gender*), informasi produk (*Product Purchased, Date of Purchase*), detail tiket (*Ticket ID, Ticket Type, Ticket Subject, Ticket Description*), status dan proses penanganan tiket (*Ticket Status, Resolution, Ticket Priority, Ticket Channel*), serta metrik layanan (*First Response Time, Time to Resolution, Customer Satisfaction Rating*), dengan jumlah keseluruhan dataset sebanyak 8.649 baris data.

Pada kolom *Ticket Priority*, dataset terbagi menjadi empat kelas dengan proporsi yang tidak seimbang, yaitu kelas *Medium* sebanyak 2.192 data, kelas *Critical* sebanyak 2.129 data, kelas *High* sebanyak 2.085 data, dan kelas *Low* sebanyak 2.063 data.



Gambar 1. Tahapan CRISP DM

Metodologi CRISP-DM (*Cross-Industry Standard Process for Data Mining*) secara umum terdiri dari enam tahapan yang terstruktur, yaitu:

2.1. Business Understanding

Tahap ini berfokus pada perumusan tujuan penelitian, yaitu mengembangkan dan membandingkan kinerja algoritma SVM dan *Logistic Regression* dalam menentukan prioritas tiket dukungan pelanggan. Pada tahap ini ditetapkan kebutuhan bisnis, variabel yang relevan (misalnya kategori masalah, urgensi tiket, waktu respons awal, dan teks deskripsi), serta kriteria evaluasi yang akan digunakan untuk menentukan model terbaik.

2.2. Data Understanding

Tahap ini mencakup eksplorasi struktur dan karakteristik dataset tiket dukungan pelanggan. Aktivitas meliputi identifikasi jumlah data, tipe variabel, proporsi setiap level prioritas (*Low, Medium, High, critical*), serta pemeriksaan adanya duplikasi, missing values, dan ketidakseimbangan kelas. Selain itu, dilakukan analisis awal terhadap

fitur teks pada deskripsi tiket untuk memahami pola-pola umum yang dapat memengaruhi penentuan prioritas.

2.3. Data Preparation

Tahap ini bertujuan untuk memastikan bahwa data berada dalam kondisi yang optimal sebelum digunakan oleh algoritma pemodelan. Proses yang dilakukan mencakup pembersihan data, penanganan missing values, encoding pada variabel kategorikal, serta normalisasi variabel numerik. Pada fitur teks, dilakukan serangkaian *text preprocessing* seperti *case folding*, tokenisasi, *stopword removal*, *stemming*, dan representasi TF-IDF. Jika ditemukan ketidakseimbangan kelas, diterapkan pula teknik penyeimbangan data seperti SMOTE. Setelah proses tersebut, data dibagi menjadi training set dan test set agar siap memasuki tahap pemodelan.

Pada tahap ini pula, kelas pada kolom *Ticket Priority* disederhanakan menjadi dua kelas biner (0 dan 1), di mana kelas 0 berisi kategori *Critical* dan *High*, sedangkan kelas 1 berisi kategori *Low* dan *Medium*. Penyederhanaan ini dilakukan untuk mengurangi kompleksitas pemodelan sekaligus meningkatkan kinerja algoritma SVM dan *Logistic Regression*, yang lebih optimal dalam skenario klasifikasi biner. Transformasi tersebut juga membantu mengatasi ketidakseimbangan distribusi kelas pada dataset asli..

2.4. Modeling

Pada tahap *Modeling*, dibangun dua model klasifikasi, yaitu *Support Vector Machine* (SVM) dan *Logistic Regression*. Kedua model dilatih menggunakan data yang telah dipersiapkan pada tahap sebelumnya untuk memperoleh performa terbaik dalam memprediksi level prioritas tiket berdasarkan fitur numerik, kategorikal, dan teks yang direpresentasikan dalam bentuk TF-IDF.

Model SVM menggunakan *kernel linear*, yang sesuai untuk data TF-IDF yang berdimensi tinggi dan cenderung dapat dipisahkan menggunakan *hyperplane linear*. Parameter regularisasi ditetapkan pada $C=1$, sehingga model mampu menyeimbangkan antara *margin* pemisahan dan tingkat kesalahan klasifikasi. Parameter gamma tidak digunakan karena tidak relevan pada kernel linear.

Model *Logistic Regression* dibangun dengan konfigurasi *multi_class = 'multinomial'*, sehingga estimasi probabilitas antar kelas dihitung secara simultan melalui pendekatan softmax. Proses optimasi menggunakan solver LBFGS, yang efisien untuk data berdimensi tinggi, serta menerapkan regularisasi L2 secara default untuk menjaga stabilitas dan generalisasi model..

2.5. Evaluation

Tahap evaluasi membandingkan performa kedua model menggunakan metrik seperti *accuracy*, *precision*, *recall*, *F1-score*, serta analisis *confusion matrix*. Evaluasi dilakukan pada data uji untuk

memastikan objektivitas performa model. Hasil evaluasi digunakan untuk mengidentifikasi model yang paling efektif dan konsisten dalam memprediksi prioritas tiket.

2.6. Deployment

Tahap ini menyajikan hasil penelitian dalam bentuk tabel, grafik, dan narasi interpretatif. Output meliputi model terbaik, analisis hasil perbandingan, serta rekomendasi penggunaan model dalam lingkungan operasional sistem dukungan pelanggan. Tahap ini memastikan hasil penelitian dapat dipahami dan diterapkan oleh praktisi maupun peneliti lain.

3. Hasil dan Pembahasan

Pembahasan hasil penelitian disusun mengikuti tahapan CRISP-DM agar selaras dengan metodologi yang digunakan, yaitu Business Understanding, Data Understanding, Data Preparation, Modeling, dan Evaluation.

3.1. Business Understanding

Pada tahap ini, permasalahan utama yang dihadapi adalah tingginya volume tiket dukungan pelanggan yang memerlukan penentuan prioritas secara cepat dan konsisten. Penentuan prioritas secara manual berpotensi menimbulkan keterlambatan penanganan tiket dengan urgensi tinggi. Oleh karena itu, penelitian ini memfokuskan pada penerapan klasifikasi berbasis teks untuk membantu proses penentuan prioritas tiket secara otomatis.

3.2. Data Understanding

Pada tahap Data Understanding, dilakukan proses eksplorasi awal terhadap dataset *customer_support_tickets.csv* untuk memahami karakteristik data yang digunakan dalam penelitian ini. Dataset ini berisi kumpulan tiket dukungan pelanggan yang dikumpulkan dari sistem helpdesk sebuah perusahaan yang menyediakan layanan teknologi. Dataset yang digunakan dalam penelitian ini terdiri dari 17 atribut yang merepresentasikan informasi pelanggan, produk yang dibeli, serta detail tiket dukungan pelanggan. Atribut tersebut meliputi *Customer ID*, *Customer Name*, *Customer Email*, dan *Customer Phone* sebagai informasi dasar pelanggan. Informasi terkait produk direpresentasikan oleh *Product Category*, *Product Purchased*, serta *Date of Purchase*.

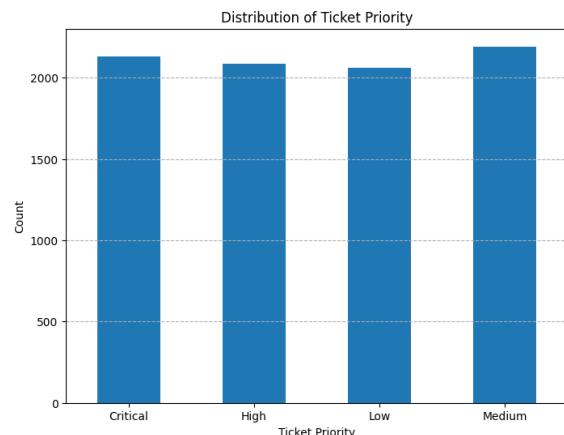
```
# Cek missing values
print(f"\n Missing values:")
print(f" - Ticket Description: {df['Ticket Description'].isnull().sum()}")
print(f" - Ticket Priority: {df['Ticket Priority'].isnull().sum()}")
```

```
Missing values:
- Ticket Description: 0
- Ticket Priority: 0
```

Gambar 2. Missing Value

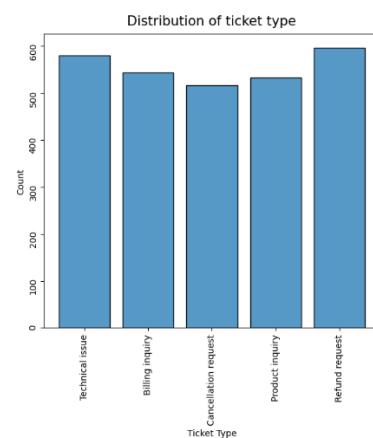
Berdasarkan hasil pemeriksaan, dataset tidak mengandung missing value (Gambar 2), sehingga seluruh atribut dapat diproses tanpa perlakuan imputasi. Atribut yang paling relevan untuk penentuan prioritas tiket adalah *Ticket Type*, *Ticket*

Subject, dan terutama *Ticket Description* yang memuat konteks permasalahan pelanggan.



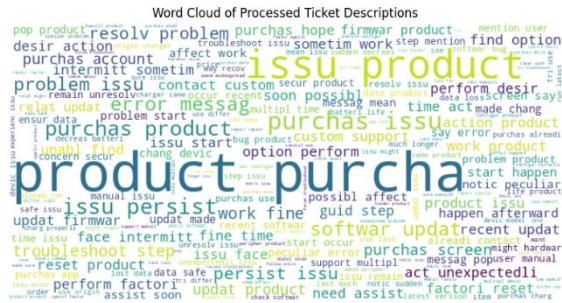
Gambar 3. Distribusi Tiket Prioritas

Berdasarkan gambar 3, Grafik tersebut menunjukkan distribusi jumlah tiket berdasarkan tingkat prioritas, yaitu *Critical*, *High*, *Medium*, dan *Low*. Secara keseluruhan, jumlah tiket pada setiap kategori berada pada kisaran yang relatif seimbang, sekitar 2000-an. Kategori *Medium* memiliki jumlah tiket terbanyak, yang mengindikasikan bahwa sebagian besar masalah yang dilaporkan bersifat cukup penting namun tidak mendesak. Sementara itu, kategori *Low* memiliki jumlah paling sedikit, meskipun tetap berada pada rentang yang tidak jauh berbeda. Adapun jumlah tiket *Critical* yang cukup tinggi menunjukkan adanya banyak kasus yang dianggap darurat atau kemungkinan kurang konsistennya penentuan prioritas. Secara umum, pola ini menggambarkan persebaran tiket yang merata dengan kecenderungan masalah moderat lebih dominan.



Gambar 4. Distribusi Ticket Type

Berdasarkan gambar 4, Grafik menunjukkan distribusi jenis tiket, di mana "*Refund request*" memiliki jumlah tiket paling tinggi (lebih dari 600), diikuti oleh "*Technical issue*" dan "*Billing inquiry*".



Gambar 5. Word Cloud Tiket Deskripsi

Berdasarkan gambar 5, *Word cloud* tersebut menggambarkan frekuensi kemunculan kata dalam deskripsi tiket yang telah melalui tahap pra-proses teks. Ukuran kata yang lebih besar menunjukkan tingkat kemunculan yang lebih tinggi, sehingga memberikan indikasi awal mengenai pola masalah yang paling sering dilaporkan. Dominasi kata seperti *product*, *purcha* (hasil *stemming* dari *purchase*), *issu*, *troubleshoot*, *error*, dan *update* menunjukkan bahwa sebagian besar tiket berkaitan dengan permasalahan produk, kendala pada proses pembelian, serta isu teknis yang memerlukan penanganan seperti *troubleshooting* dan pembaruan sistem. Visualisasi ini membantu mengidentifikasi fokus utama keluhan pengguna secara cepat dan berfungsi sebagai dasar untuk analisis lebih mendalam mengenai kategori masalah yang muncul dalam sistem dukungan pelanggan.

3.3. Data Preparation

3.3.1 Text Preprocessing

Preprocessing merupakan tahap pengolahan data yang telah dikumpulkan menjadi satu dokumen, yang kemudian akan menjalani proses analisis. Preprocessing ini terbagi menjadi beberapa bagian, yaitu:

1. Lowercasing

Seluruh teks diubah menjadi huruf kecil untuk memastikan konsistensi dan menghindari perbedaan kata yang hanya disebabkan oleh kapitalisasi.

2. Pembersihan Karakter (*Cleaning*)

Dilakukan penghapusan karakter yang tidak relevan seperti angka, tanda baca, dan simbol menggunakan regular expression. Hanya huruf dan spasi yang dipertahankan. Selain itu, spasi berlebihan juga dihilangkan agar format teks menjadi lebih rapi.

3. Tokenisasi

Teks yang telah dibersihkan kemudian dipecah menjadi token (kata-kata) menggunakan word tokenizer.

4. *Stopwords Removal* dengan *Stopwords Tambahan* Stopwords standar dari NLTK digunakan untuk menghapus kata-kata umum yang tidak memiliki kontribusi terhadap pemodelan. Selain itu, ditambahkan pula stopwords khusus yang sering

muncul dalam konteks tiket dukungan pelanggan seperti:

“please”, “assist”, “help”, “thank”, “thanks”,
“hello”, “hi”, “dear”, “regards”.

karena kata-kata ini tidak memberikan informasi terkait permasalahan yang dihadapi pelanggan.

5. Penghapusan Token Tidak Relevan

Token dengan panjang kurang dari tiga karakter dihapus karena biasanya tidak memiliki makna penting dalam analisis teks.

6. Stemming

Setiap token dikonversi ke bentuk dasarnya menggunakan Porter Stemmer. Proses ini membantu menyatukan variasi kata yang memiliki makna sama, misalnya “running”, “runs”, dan “run” menjadi “run”.

```
# 1. CLEANING
# Convert to lowercase
text = str(text).lower()

# Remove special characters, numbers, but keep meaningful symbols
text = re.sub(r'[^a-zA-Z\s]', ' ', text)

# Remove extra whitespace
text = re.sub(r'\s+', ' ', text).strip()

# 2. TOKENIZING
tokens = word_tokenize(text)

# 3. STOPWORD REMOVAL
tokens = [token for token in tokens
          if token not in stop_words and len(token) > 2]

# 4. STEMMING
stemmed_tokens = [stemmer.stem(token) for token in tokens]

# Join kembali menjadi teks
processed_text = ''.join(stemmed_tokens)
```

Gambar 6. Pre Processing

3.3.2 Transformasi Teks Menggunakan TF-IDF

Setelah proses pembersihan, teks dikonversi menjadi representasi numerik menggunakan TF-IDF (*Term Frequency – Inverse Document Frequency*). Metode ini digunakan karena mampu memberikan bobot lebih tinggi pada kata-kata penting dalam dokumen, serta mengurangi pengaruh kata yang terlalu sering muncul. TF-IDF digunakan untuk mentransformasikan fitur teks *Ticket Description* menjadi vektor numerik berdimensi tinggi yang dapat diproses oleh algoritma SVM dan *Logistic Regression*.

```

# Inisialisasi TF-IDF Vectorizer dengan max_features dan ngram_range
max_features: jumlah kumpulan term yang berada dalam dataset
ngram_range=(1, 2): menggunakan unigram (kata tunggal) dan bigram (dua kata berurutan)
print("IMPLEMENTASI TF-IDF VECTORIZATOR DENGAN N-GRAM...")
tfidf_vectorizer = TfidfVectorizer(max_features=1000, ngram_range=(1, 2))

MELAKUKAN TF-IDF VECTORIZATOR DENGAN N-GRAM...
# Mengakses TF-IDF pada kolom "Ticket Description_processed"
X_tfidf = tfidf_vectorizer.fit_transform(df['Ticket Description_processed']).toarray()

# Konversi numpy array ke Dataframe
X_tfidf_df = pd.DataFrame(X_tfidf, columns=tfidf_vectorizer.get_feature_names_out())
print("Ukuran matriks TF-IDF: ", X_tfidf_df.shape)

Ukuran matriks TF-IDF: (8449, 1000)

```

Gambar 7. TF IDF

3.3.3 Encoding Label Target

Atribut *Ticket Priority*, yang merupakan variabel target, diubah menjadi nilai numerik menggunakan label encoding. Proses ini diperlukan agar nilai kategorikal seperti *low*, *medium*, *high*, dan *critical* dapat diproses oleh model klasifikasi. Pada proses

encoding ini, kelas *critical* dan *high* dikonversi menjadi nilai 0, sedangkan kelas *low* dan *medium* dikonversi menjadi nilai 1, sehingga target dapat digunakan dalam pemodelan biner.

3.3.4 Pembagian Dataset (*Train–Test Split*)

Pada tahap pembagian data, proses data splitting dilakukan dengan proporsi berbeda untuk masing-masing model guna memperoleh performa terbaik dari setiap algoritma. Untuk *Support Vector Machine* (SVM), data dibagi dengan rasio 80% data latih dan 20% data uji, sedangkan pada *Logistic Regression* digunakan rasio 70% data latih dan 30% data uji. Perbedaan rasio ini dipilih karena masing-masing konfigurasi menghasilkan nilai akurasi yang paling optimal untuk model terkait.

3.4 Modeling

Tahap Modeling dilakukan dengan menerapkan dua algoritma klasifikasi, yaitu Support Vector Machine (SVM) dengan kernel linear dan Logistic Regression. Dataset dibagi menjadi data latih dan data uji menggunakan rasio 80:20. Kedua model dilatih menggunakan fitur TF-IDF hasil tahap sebelumnya dan dikonfigurasikan untuk tugas klasifikasi biner.

3.5 Evaluation

Evaluasi performa model dilakukan menggunakan metrik Accuracy, Precision, Recall, dan F1-score. Tabel 1 dan Tabel 2 menyajikan hasil evaluasi masing-masing model.

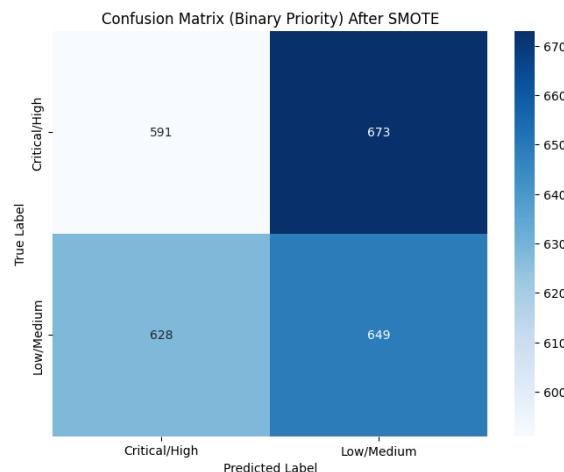
Tabel 1 Hasil Evaluasi Model Support Vector Machine (SVM)

Metrik	Nilai
Accuracy	48,80%
Precision	0,488
Recall	0,488
F1-score	0,488

Tabel 2 Hasil Evaluasi Model Logistic Regression

Metrik	Nilai
Accuracy	51,33%
Precision	0,511
Recall	0,513
F1-score	0,508

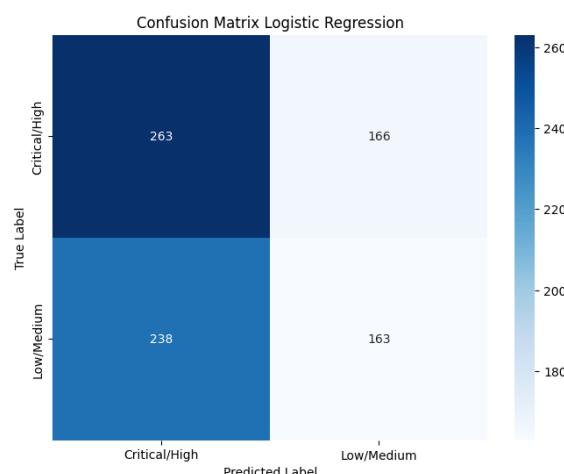
Untuk memberikan gambaran yang lebih rinci mengenai kemampuan model dalam mengklasifikasikan prioritas tiket, confusion matrix dari masing-masing model disajikan pada gambar 3 dan gambar 4.



Gambar 8. Confusion Matrix Model SVM

Tabel 3. Confusion Matrix SVM

	Predicted Critical/High	Predicted Low/Medium	Total Actual
Critical/High	591 (True Positive)	673 (False Negative)	1.264
Low/Medium	628 (False Positive)	694 (True Negative)	12.77
Total predicted	1.219	1.322	2.541



Gambar 9 Hasil Evaluasi Model Logistic Regression

Tabel 4. Confusion Matrix Logistic Regression

	Predicted Critical/High	Predicted Low/Medium	Total Actual
Critical/High	263 (True Positive)	166 (False Negative)	429

Low/Medium	238(False Positive)	163 (True Negative)	401
Total predicted	501	329	830

Berdasarkan hasil evaluasi, kedua model menunjukkan performa yang relatif sederhana dan belum optimal. Model Logistic Regression memiliki akurasi sedikit lebih tinggi dibandingkan SVM, namun menunjukkan ketidakseimbangan performa antar kelas, terutama pada kelas prioritas tinggi yang memiliki nilai recall lebih rendah. Hal ini mengindikasikan bahwa model masih kesulitan dalam membedakan urgensi tiket secara konsisten.

Rendahnya performa kedua model dapat dijelaskan oleh beberapa faktor utama. Pertama, tingginya kemiripan bahasa pada deskripsi tiket antar tingkat prioritas menyebabkan fitur yang dihasilkan sulit dibedakan secara linear. Kedua, tidak digunakan fitur non-teks seperti jenis masalah, waktu pelaporan, atau riwayat pelanggan membatasi konteks yang dapat dipelajari oleh model. Ketiga, metode TF-IDF tidak mampu menangkap konteks dan urgensi secara semantik, karena hanya merepresentasikan frekuensi kemunculan kata tanpa memahami makna kalimat secara utuh.

Temuan dalam penelitian ini terbukti konsisten dengan berbagai studi terdahulu yang menunjukkan bahwa pendekatan klasifikasi teks murni menggunakan algoritma klasik sering kali menghasilkan performa rendah pada tugas penentuan urgensi tiket. Studi lain melaporkan bahwa ketika deskripsi masalah bersifat umum dan homogen, metode TF-IDF dan algoritma seperti SVM atau Logistic Regression sulit mencapai akurasi tinggi karena keterbatasan dalam menangkap nuansa bahasa. Meskipun dalam beberapa domain seperti analisis sentimen algoritma ini bisa mencapai akurasi di atas 80%, efektivitasnya sangat bergantung pada struktur data dan tingkat pemisahan antar kelas yang ada.

Sebagai alternatif solusi, penelitian selanjutnya dapat mengeksplorasi penggunaan word embedding seperti Word2Vec, GloVe, atau FastText untuk menangkap hubungan semantik antar kata. Selain itu, pendekatan deep learning berbasis Transformer seperti BERT berpotensi meningkatkan performa karena mampu memahami konteks kalimat secara lebih mendalam. Integrasi metadata non-teks serta penerapan ensemble model juga dapat dipertimbangkan untuk meningkatkan akurasi dan stabilitas klasifikasi prioritas tiket.

3.6 Deployment

Pada tahap deployment, penelitian ini tidak melakukan implementasi model secara operasional, namun menyusun rancangan penerapan yang dapat digunakan pada lingkungan produksi. Rancangan ini meliputi penyimpanan artefak model, termasuk

pipeline *preprocessing*, TF-IDF vectorizer, serta model SVM dan *Logistic Regression* yang telah dilatih. Model kemudian diusulkan untuk diintegrasikan ke dalam sistem tiket dukungan pelanggan melalui API prediksi yang memproses input deskripsi tiket secara otomatis dan menghasilkan estimasi prioritas. Selain itu, direncanakan pula mekanisme pemantauan performa model, logging prediksi, serta prosedur *retraining* berkala untuk mengantisipasi perubahan pola data (*concept drift*). Meskipun implementasi penuh belum dilakukan, tahap *deployment* ini menyediakan kerangka teknis yang memungkinkan model untuk diterapkan secara nyata dan mendukung keputusan penentuan prioritas tiket di lingkungan operasional.

4. Kesimpulan

Berdasarkan analisis perbandingan kinerja algoritma dalam penentuan prioritas tiket dukungan pelanggan, disimpulkan bahwa kedua model yang dikembangkan belum menunjukkan performa yang optimal. Algoritma *Logistic Regression* terbukti sedikit lebih unggul dengan pencapaian akurasi sebesar 51,3%, dibandingkan dengan *Support Vector Machine* (SVM) yang menghasilkan akurasi 50,3%. Rendahnya tingkat akurasi yang hanya sedikit di atas prediksi acak ini mengindikasikan bahwa fitur teks deskripsi yang diekstraksi menggunakan metode TF-IDF semata tidak cukup kuat untuk membedakan tingkat urgensi tiket secara signifikan. Hal ini menunjukkan pola kata dalam deskripsi keluhan pelanggan memiliki kemiripan yang tinggi antar tingkat prioritas, sehingga sulit dipisahkan secara linear oleh kedua algoritma tersebut. Oleh karena itu, penerapan model ini secara langsung untuk otomatisasi operasional belum direkomendasikan tanpa adanya pengembangan lebih lanjut. Untuk penelitian selanjutnya, disarankan untuk mengeksplorasi teknik ekstraksi fitur yang lebih kompleks seperti *word embedding* atau pendekatan *deep learning* guna menangkap konteks semantik teks dengan lebih baik, serta mengintegrasikan fitur non-teks (metadata) untuk meningkatkan akurasi klasifikasi.

Daftar Rujukan

- [1] S. Fuchs, C. Drieschner, and H. Wittges, “Improving Support Ticket Systems Using Machine Learning: A Literature Review,” in *Proceedings of the 55th Hawaii International Conference on System Sciences*, Jan. 2022, pp. 1893–1902. doi: <https://scholarspace.manoa.hawaii.edu/10.24251/HICSS.2022.238>.
- [2] A. Muzukasi, C. Rubino, E. Sutanto, and G. Anandha, “Menavigasi Dinamika Industri Hiburan: Strategi Inovatif PT EKIS MUDA BERKARYA dalam Menjawab Tantangan Event dan Ticketing Digital,” *Ekopedia: Jurnal Ilmiah Ekonomi*, vol. 1, Jun. 2025.
- [3] D. Franzely *et al.*, “Sistem Pendukung Keputusan Penentuan Prioritas Pengerjaan Order Berdasarkan

- Penilaian dan Pengelompokan Pelanggan Pada PT Absolute Akustik Indonesia,” 2024.
- [4] A. Hermawan, I. Jowensen, J. Junaedi, and Edy, “Implementasi Text-Mining untuk Analisis Sentimen pada Twitter dengan Algoritma Support Vector Machine,” *JST (Jurnal Sains dan Teknologi)*, vol. 12, no. 1, pp. 129–137, Apr. 2023, doi: 10.23887/jstundiksha.v12i1.52358.
- [5] A. R. Hanum *et al.*, “Analisis Kinerja Algoritma Klasifikasi Teks BERT Dalam Mendeteksi Berita Hoaks,” vol. 11, no. 3, pp. 537–546, 2024, doi: 10.25126/jtiik938093.
- [6] A. Desiani, D. A. Zayanti, I. Ramayanti, F. F. Ramadhan, and Giovillando, “Perbandingan Algoritma Support Vector Machine (SVM) dan Logistic Regression Dalam Klasifikasi Kanker Payudara,” *Jurnal Kecerdasan Buatan dan Teknologi Informasi*, vol. 4, no. 1, pp. 33–42, Jan. 2025, doi: 10.69916/jkbt.v4i1.191.
- [7] S. Rabbani, D. Safitri, N. Rahmadhani, A. A. F. Sani, and M. K. Anam, “Perbandingan Evaluasi Kernel SVM untuk Klasifikasi Sentimen dalam Analisis Kenaikan Harga BBM,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 3, no. 2, pp. 153–160, Oct. 2023, doi: 10.57152/malcom.v3i2.897.
- [8] O. S. D. Fadhillah, J. H. Jaman, and C. Carudin, “PERBANDINGAN NAIVE BAYES, SUPPORT VECTOR MACHINE, LOGISTIC REGRESSION DAN RANDOM FOREST DALAM MENGANALISIS SENTIMEN MENGENAI TIKTOKSHOP,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 1, pp. 840–847, Jan. 2025, doi: 10.23960/jitet.v13i1.5746.
- [9] S. G. Alengka, J. L. Putra, and T. Setiyorini, “Analisis Sentimen Ulasan Mobile Legend Menggunakan Algoritma Naive Bayes, SVM, Logistic Regression,” *Jurnal Algoritme*, vol. 5, no. 3, pp. 439–449, 2025, doi: 10.35957/algoritme.v5i3.12915.
- [10] R. A. KHOMEINI NOOR BINTANG and N. T. Romadloni, “Perbandingan Kinerja Algoritma Klasifikasi Pada Review Pengguna Aplikasi Netflix,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 2, Apr. 2025, doi: 10.23960/jitet.v13i2.6303.
- [11] K. W. Gusti, “Perbandingan Metode Support Vector Machine dan Logistic Regression Untuk Klasifikasi Bencana Alam,” Aug. 2023, [Online]. Available: <https://t.co/JwC9D8xnKe>
- [12] V. Agresia and R. R. Suryono, “Comparison of SVM, Naive Bayes, and Logistic Regression Algorithms for Sentiment Analysis of Fraud and Bots In Purchasing Concert Ticket,” vol. 10, no. 2, p. 2025.
- [13] I. Syahrohim, S. D. Saputra, R. W. Saputra, V. H. Pranatawijaya, and R. Priskila, “Perbandingan Analisis Sentimen Setelah Pilpres 2024 di Twitter Menggunakan Algoritma Machine Learning,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 2, Apr. 2024, doi: 10.23960/jitet.v12i2.4249.
- [14] A. G. Budianto, R. Rusilawati, A. T. E. Suryo, G. R. Cahyono, A. F. Zulkarnain, and M. Martunus, “Perbandingan Performa Algoritma Support Vector Machine (SVM) dan Logistic Regression untuk Analisis Sentimen Pengguna Aplikasi Retail di Android,” *Jurnal Sains dan Informatika*, vol. 10, no. 2, Nov. 2024, doi: 10.34128/jsi.v10i2.911.
- [15] K. Subham, “NLP-Based Automation in Customer Support and Case Management,” *International journal of networks and security*, vol. 05, no. 01, pp. 91–117, May 2025, doi: 10.55640/ijns-05-01-07.
- [16] D. Kurniawan and D. M. Yasir, “Optimization Sentiment Analysis Using CRISP-DM and NAÏVE BAYES Methos Implemented on Social Media,” *Cyberpace: Jurnal Pendidikan Teknologi informasi*, vol. 6, Oct. 2022.
- [17] I. Mustofa, A. H. Wibowo, K. A. Sekarjati, N. S. Makbulina, and R. Dewangga, “Penerapan Association Rule-Market Basket Analysis (AR-MBA) Dalam Menentukan Strategi Product Bundling: Studi Kasus Pada Minimarket AKPRIND MART,” *Jurnal Teknik Industri Terintegrasi*, vol. 7, no. 1, pp. 379–386, Jan. 2024, doi: 10.31004/jutin.v7i1.24873.
- [18] N. Hidayati, J. Suntoro, and G. G. Setiaji, “Perbandingan Algoritma Klasifikasi untuk Prediksi Cacat Software dengan Pendekatan CRISP-DM,” *Jurnal Sains dan Informatika*, vol. 7, no. 2, pp. 117–126, Nov. 2021, doi: 10.34128/jsi.v7i2.313.