khazanah informatika

Jurnal Ilmu Komputer dan Informatika

Hyperparameter Optimization of TF-IDF and SVM via Grid Search for Sentiment Analysis of Traveloka Customer Reviews

Muhammad Bayu Kurniawan [1*], Hanafi [2], Riki Hikmianto [3], Isnawati Muslihah [4]

*bayu.x@students.amikom.ac.id

[1,2,3]Department Master of Informatics
Universitas AMIKOM Yogyakarta
Yogyakarta, Indonesia

[4]Department of Art and Design
Institut Seni Indonesia Surakarta

Abstract-Customer reviews on digital platforms are crucial for improving services and making business decisions. This study focuses on automated sentiment analysis for Traveloka, a leading Indonesian online travel application. We propose a systematic hyperparameter optimization of a combined TF-IDF and Support Vector Machine (SVM) pipeline. A dataset of 20,200 user reviews was collected from the Google Play Store. After preprocessing and a two-stage labeling process, the data was split using stratified sampling (70% training, 30% testing). We conducted a comprehensive Grid Search with stratified 5-fold cross-validation to jointly optimize TF-IDF n-gram ranges (unigram, bigram, trigram) and SVM hyperparameters across four kernel types (Linear, RBF, Polynomial, Sigmoid). The results show that the Polynomial kernel with trigram features (C=5, gamma=1, degree=5, coef0=10) performs best. It achieves a test accuracy of 87.10% and a macro F1-score of 86.9%. Error analysis revealed the model's high reliability in detecting negative feedback (precision: 90.4%) but also its difficulty with contrastive sentences and informal language. The minimal performance differences among top configurations suggest the task is robust to specific parameter choices. However, the model's bag-of-ngrams approach shows limitations in processing contrastive sentences and informal language. For future work, employing contextual embeddings (e.g., IndoBERT) and exploring alternative algorithms like Random Forest or Neural Networks could address these challenges. This research presents a thoroughly optimized traditional ML methodology that establishes a strong baseline for automated sentiment analysis of Indonesian user feedback.

Keywords: Sentiment Analysis, Hyperparameter Optimization, Grid Search, Support Vector Machine, TF-IDF, Traveloka.

Article info: *submitted April 23, 2025, revised January 23, 2026, accepted January 26, 2026*

1. Introduction

The proliferation of internet users has fundamentally reshaped various business sectors, notably the travel and lifestyle industry [1]. In Indonesia, survey data from the Indonesian Internet Service Providers Association (APJII) indicates a continued rise in internet penetration, growing by 1.16% in 2025 [2]. This digital expansion has increased societal reliance on online platforms for travel planning and accommodation booking, catalyzing the rapid growth of review-based travel applications [3]. These platforms generate vast amounts of user-generated content (UGC), comprising diverse customer feedback that holds strategic value for companies seeking to enhance service quality through sentiment analysis [4]. As a leading travel platform in Southeast Asia, with over 50 million downloads and 2 million reviews contributing to a 4.8-star rating on the Google Play Store [5], Traveloka presents an ideal case study for extracting sentiment-based insights. Analyzing such reviews enables service monitoring, issue identification, and data-driven improvements, but the volume and complexity of this data require reliable automated classification methods [6].

The global trajectory of sentiment analysis research shows a marked shift towards deep learning architectures, such as BERT Transformers, and RoBERTa, which excel at capturing complex semantic and contextual nuances [7]. Nonetheless, these models present significant practical barriers, including demands for substantial computational resources, considerable amounts of task-specific labeled data for effective fine-tuning, and intricate, time-consuming training processes [8]. Their inherent "black-box" nature also poses challenges for interpretability, which is often crucial in business applications requiring transparency and actionable insight. Consequently, for medium-scale datasets or resource-constrained environments, traditional machine learning algorithms like Support Vector Machine (SVM) remain a highly relevant and efficient alternative. SVM offers a compelling balance of computational efficiency, model interpretability, and robust classification performance, particularly on structured or semi-structured text data [9], [10].

The efficacy of SVM is well-documented, especially when paired with the Term Frequency–Inverse Document Frequency (TF-IDF) feature extraction method. Prior research consistently demonstrates its strong performance: a study on the MySAPK application (4,778 reviews) reported 94.14% accuracy for SVM, surpassing Naïve Bayes [11]; analysis of Neobank app reviews (3,159 samples) achieved 82.33% accuracy [12]; and work on the TikTok app (8,785 reviews) yielded an 80% F1-score [13]. Further validation comes from studies on the by.U application, where SVM with TF-IDF achieved 84.7% average accuracy using 5-fold cross-validation [14], and on hotel review in Jordan, where SVM with

TF-IDF outperformed several other classifiers, achieving 97% accuracy [15]. Research by Muhammadi et al. [16] on Traveloka reviews, combining lexicon-based labeling with TF-IDF, found the SVM RBF kernel achieved 81.73% accuracy. Similarly, Agustina et al. [17] reported that TF-IDF with an SVM-RBF model attained precision, recall, and F1-score of 85%, 86%, and 84%, respectively, for vaccine sentiment analysis, outperforming other kernels.

Collectively, these studies affirm the TF-IDF and SVM pipeline as a robust baseline for sentiment classification. However, its performance is critically dependent on the careful selection of hyperparameters, a step often underrepresented or oversimplified in existing literature [11-17]. This study identifies two specific and interconnected research gaps: First, many these prior studies (e.g., [12], [14], [15]) rely on default hyperparameters or limited optimization for SVM, despite its sensitivity to kernel choice, regularization parameter (C), and kernel-specific parameters like gamma. Second, the systematic exploration of the ngram_range parameter in TF-IDF is frequently neglected [11-17], even though it fundamentally shapes the contextual features presented to the classifier and can drastically impact results. Comprehensive hyperparameter optimization across both components presents a cost-effective strategy to maximize accuracy without altering the core model architecture [18].

To address these gaps methodically, this research employs the Grid Search method. Grid Search provides a systematic, exhaustive, and reproducible framework for exploring a defined hyperparameter space [19]. While computationally more intensive than random search, its exhaustive nature is justified here as it guarantees the identification of the optimal combination within predefined bounds and enables a complete sensitivity analysis; a core objective of this investigative study [20]. To ensure a robust evaluation that mitigates overfitting and accounts for potential class imbalance in the review dataset, the optimization process is validated using stratified k-fold cross-validation [21]. By bridging these identified gaps, this study offers a comprehensive and systematic framework for the joint hyperparameter tuning of the TF-IDF and SVM pipeline. Its primary scientific contribution is the delivery of empirical evidence and clear guidelines on the optimal interplay between textual feature scope (`ngram_range`) and SVM kernel parameters for sentiment analysis in the travel application domain. This research has three primary objectives:

1. To systematically optimize the joint hyperparameters of TF-IDF (`ngram_range`) and SVM (`kernel`, `C`, `gamma`) using Grid Search with stratified 5-fold cross-validation on a dataset of Traveloka user reviews.
2. To evaluate the performance of the optimized model using comprehensive metrics (accuracy, precision, recall, F1-score) and compare it against baseline configurations.
3. To identify and validate the optimal hyperparameter configuration for sentiment classification in the travel application review domain.

2. Methods

This study utilizes a quantitative approach and an experimental method to evaluate the performance of sentiment classification models. The research methodology comprises data collection, feature selection, data labeling, data preprocessing, TF-IDF-based feature extraction with varying n-gram ranges, implementation of the SVM algorithm with different kernels, and joint hyperparameter optimization of both TF-IDF and SVM via Grid Search with stratified 5-fold cross-validation. A comprehensive evaluation of the model was performed using a confusion matrix to obtain the accuracy, precision, recall, and F1-score metrics, ensuring reliable performance measurement. Figure 1 depicts the overall research workflow.

2.1. Data collection

The primary dataset comprises 20,200 user reviews of the Traveloka application, scraped from the Google Play Store to ensure authentic user-generated feedback. Data extraction was performed using the google-play-scraper Python library within a Google Colaboratory environment. The scraping parameters were set to: Indonesian language (id), Indonesia geographic scope (ID), sort order by 'most relevant' to capture high-engagement content, and inclusion of all star ratings (1–5) to prevent selection bias. Each data point includes metadata such as review text, Score, reviewId, userName, thumbsUpCount, reviewCreatedVersion, at, replyContent, repliedAt, and appVersion. The raw JSON data was converted into a structured pandas DataFrame for analysis. All data is publicly available, and collection adhered to the platform's terms of service and ethical research standards.

2.2. Feature Selection

Feature selection involves identifying the most relevant attributes for sentiment analysis from the dataset's metadata. While the raw data contained various columns such as userName, thumbsUpCount, and replyContent, this study specifically retains only the review text ('content') and star rating ('score') for model development. The score serves as the foundation for sentiment labeling, while the content provides the textual data for feature extraction. Other metadata were excluded to concentrate the model's learning on linguistic patterns alone, as incorporating additional non-textual features could introduce confounding variables not directly related to textual sentiment expression. The structure of the selected features is displayed in Table 1.

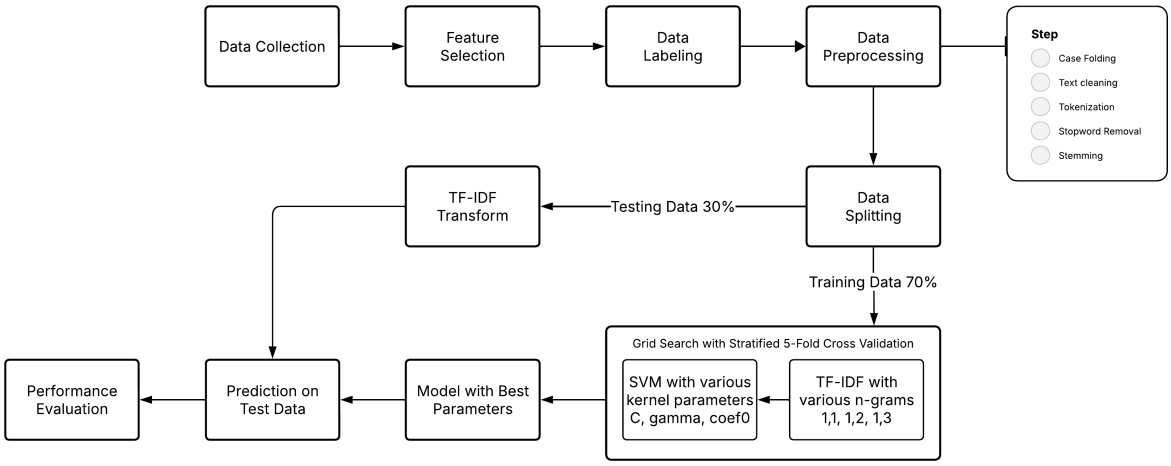

Figure 1. The flow of the research proces

Table 1. Feature selection result

Content	Score
Kenapa sekarang Traveloka jadi sering pembatalan penerbangan secara sepihak? Saya sering di rugikan, refund atau reschedule tetap merugikan saya. Tolong di perbaiki!	1
Di hp sulit sekali untuk merubah jumlah penumpang, tidak bergulir	3
Terimakasih Traveloka, Dengan Traveloka saya selalu bisa pesan tiket untuk traveling. Dan customer service nya selalu membalas keluhan pelanggan nya. 👆	5

2.3. Data Labeling & Validation

Sentiment labels were assigned through a two-stage process to ensure label reliability. First, an automatic rule-based labeling was applied where reviews with scores of 1-3 were labeled as 'Negative' and scores of 4-5 as 'Positive'. This approach is common in review-based sentiment analysis. To enhance label accuracy, a targeted manual verification was subsequently conducted. Reviews with ambiguous scores (2, 3, and 4 stars) were prioritized for examination, as these ratings most frequently exhibit sentiment-text dissonance. The research team manually reviewed and corrected labels for these ambiguous cases based solely on textual sentiment. Additionally, a random sample of 1-star and 5-star reviews was spot-checked for consistency. This verification process refined the dataset by correcting clear mismatches (e.g., a review stating "sangat puas" with a 2-star rating) before model training and evaluation. The final verified labels were used as ground truth, as shown in Table 2.

Table 2. Labeling result

Content	Score	Label
Kenapa sekarang Traveloka jadi sering pembatalan penerbangan secara sepihak? Saya sering di rugikan, refund atau reschedule tetap merugikan saya. Tolong di perbaiki!	1	Negative
Di hp sulit sekali untuk merubah jumlah penumpang, tidak bergulir	3	Negative
Terimakasih Traveloka, Dengan Traveloka saya selalu bisa pesan tiket untuk traveling. Dan customer service nya selalu membalas keluhan pelanggan nya. 👆	5	Positive

2.4. Data Preprocessing

Text preprocessing standardizes and refines raw review text to enable effective feature extraction and model training [22]. This study implements a five-stage pipeline optimized for informal Indonesian user-generated reviews: case folding, text cleaning, tokenization, stopword removal, and stemming. Table 3 illustrates the complete transformation process applied to a sample review.

1. Case Folding : The initial step converts all characters to lowercase to ensure consistency and prevent the same word in different cases (e.g., "Traveloka" vs. "traveloka") from being treated as distinct features [23]. This normalization reduces the dimensionality of the feature space.
2. Text Cleaning : The uniform lowercase text is then cleansed of non-linguistic elements that do not carry sentiment, such as URLs, punctuation, special characters, emoticons, and numerical digits [24]. This process isolates the core textual content and removes noise that could hinder accurate analysis.
3. Tokenization : Subsequently, the cleaned text is segmented into individual word units, or tokens, based on whitespace and delimiters [25]. Tokenization breaks down the text into its fundamental components, enabling frequency analysis and feature extraction for the machine learning model.
4. Stopword Removal : High-frequency grammatical words with minimal semantic value for sentiment (e.g., "dan", "yang", "di", "the", "and") are removed using a predefined stopword list from the Natural Language Toolkit (NLTK) library for Indonesian [26]. This step filters out lexical noise, further refines the feature set, and improves computational efficiency.
5. Stemming : Finally, each token is reduced to its base or root form using the Sastrawi library, which is specifically designed for the Indonesian language [27]. For instance, words like "membalas" (to reply) and "pelanggan" (customer) are stemmed to "balas" and "langgan". This morphological normalization consolidates different word inflections, addressing data sparsity and enhancing the model's ability to recognize core terms.

Each step incrementally refines the data, ensuring that only the most informative linguistic features are retained for the

subsequent sentiment classification task. A complete, step-by-step illustration of this preprocessing pipeline applied to a sample review is presented in Table 3.

Table 3. Example of the Text Preprocessing Pipeline

Step	Output
Raw Text	Terimakasih Traveloka, Dengan Traveloka saya selalu bisa pesan tiket untuk traveling. Dan customer service nya selalu membalas keluhan pelanggan nya. 👍
Case Folding	terimakasih traveloka, dengan traveloka saya selalu bisa pesan tiket untuk traveling. dan customer service nya selalu membalas keluhan pelanggan nya. 👍
Text Cleaning	terimakasih traveloka dengan traveloka saya selalu bisa pesan tiket untuk traveling dan customer service nya selalu membalas keluhan pelanggan nya
Tokenization	['terimakasih', 'traveloka', 'dengan', 'traveloka', 'saya', 'selalu', 'bisa', 'pesan', 'tiket', 'untuk', 'traveling', 'dan', 'customer', 'service', 'nya', 'selalu', 'membalas', 'keluhan', 'pelanggan', 'nya']
Stopword Removal	['terimakasih', 'traveloka', 'traveloka', 'selalu', 'bisa', 'pesan', 'tiket', 'traveling', 'customer', 'service', 'selalu', 'membalas', 'keluhan', 'pelanggan']
Stemming	['terimakasih', 'traveloka', 'traveloka', 'selalu', 'bisa', 'pesan', 'tiket', 'travel', 'customer', 'service', 'selalu', 'balas', 'keluh', 'langgan']

2.5. Feature Extraction using TF-IDF

This study utilized the Term Frequency-Inverse Document Frequency (TF-IDF) method to transform the preprocessed text documents into a structured numerical vector space suitable for machine learning algorithms [28], [29]. To capture not only unigrams (single words) but also sequential word patterns, the TF-IDF vectors were constructed using a combination of n-grams (unigram, bigram, and trigram). This approach allows the model to consider phrases and multi-word expressions (e.g., "excellent service" or "not as expected"), which are often critical for accurate sentiment interpretation in reviews.

The TF-IDF weight for each n-gram feature is computed as the product of two components: Term Frequency (TF) and Inverse Document Frequency (IDF). We used the logarithmic TF scheme to normalize term counts within documents, calculated as:

$$TF(t,d) = 1 + \log(f_{t,d}) \quad (1)$$

where $f_{t,d}$ denotes the frequency of n-gram t in document d.

The IDF component assigns higher weight to rare, discriminative n-grams across the corpus, calculated as:

$$IDF(t) = 1 + \log\left(\frac{N_d}{df_t}\right) \quad (2)$$

where N_d is the total number of documents in the corpus, and df_t is the document frequency of n-gram t.

The final TF-IDF weight is given by:

$$TF.IDF(t,d) = TF(t,d).IDF(t) \quad (3)$$

In implementation, the TF-IDF vectorization was configured to generate features from unigrams, bigrams, and trigrams (n_range = 1,3). This n-gram enriched feature matrix subsequently served as the primary input for training and evaluating the sentiment classification models.

2.6. Support Vector Machine

Support Vector Machine (SVM), introduced by Vapnik, is a robust supervised learning algorithm widely used for classification tasks, including sentiment analysis [30]. Fundamentally, SVM aims to find an optimal hyperplane in a high-dimensional feature space that maximally separates data points of different classes [31]. The optimization process maximizes the margin, which is the distance between the hyperplane and the nearest data points from each class (support vectors), thereby enhancing the model's generalization ability [32].

A key strength of SVM is its flexibility in handling both linear and non-linear data through the use of kernel functions [33]. Kernels implicitly map the input features into a higher-dimensional space where a linear separation becomes possible. In this study, we evaluated the performance of four distinct kernel functions:

1. Linear Kernel: Suitable for linearly separable data.
2. Polynomial Kernel: Captures non-linear, feature relationships through polynomial combinations.
3. Radial Basis Function (RBF) Kernel: A powerful non-linear kernel that handles complex, non-linear class boundaries.
4. Sigmoid Kernel: Mimics a neural network-like structure for non-linear separation.

Compared to other classifiers, SVM is particularly effective for scenarios with high-dimensional data (such as text represented by TF-IDF vectors), smaller sample sizes, and complex non-linear relationships [34]. While it guarantees a global optimum and offers fast prediction times, its training time and space complexity can increase with larger datasets [35], [36]. In this research, the SVM algorithm was applied to the TF-IDF transformed feature set (including unigram, bigram, and trigram features) to classify customer reviews into positive and negative sentiment categories.

2.7. Hyperparameter Optimization using Grid Search

Hyperparameter optimization was conducted using Grid Search to systematically identify the optimal configuration for the combined TF-IDF and SVM pipeline. The search simultaneously explored key parameters from both components: : the n-gram range (ngram_range) for TF-IDF (e.g., (1,1), (1,2), (1,3)) and the kernel type, regularization parameter (C: e.g., 0.1, 1, 10, 100), and kernel coefficient (gamma: e.g., 0.001, 0.01, 0.1, 'scale') for SVM classification. Model evaluation employed stratified 5-fold cross-validation to ensure robust performance estimation, with the macro F1-score serving as the primary selection metric. The exhaustive Grid Search process, implemented via scikit-learn's GridSearchCV, evaluated all possible parameter combinations within the defined search space to guarantee identification of the optimal hyperparameter set for sentiment classification.

2.8. Evaluation Model

This stage evaluates the performance of the optimized SVM model in classifying sentiment (positive and negative) from Traveloka application reviews. The primary evaluation tool is the confusion matrix, a standard performance measurement for classification models that provides a detailed breakdown of correct and incorrect predictions across all classes [37]. The matrix cross-tabulates the actual labels with the model's predicted labels, generating counts for True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN), as illustrated in Table 4 [38].

Table 4. Confusion Matrix Structure

	Actual Positive	Actual Negative
Predicted Positive	TP	FP
Predicted Negative	FN	TN

From the confusion matrix, four complementary performance metrics are derived:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

$$F1\ Score = 2\ x \frac{(Recall\ x\ Precision)}{(Recall + Precision)} \quad (7)$$

In the context of sentiment analysis, precision and recall are particularly critical for understanding the model's reliability in identifying each specific sentiment class, while the F1-score provides a balanced measure that accounts for potential class imbalance.

3. Result

3.1. Dataset

The final dataset consisted of 20,200 reviews from the Traveloka application after undergoing preprocessing and manual validation. The class distribution showed a moderate imbalance, where negative reviews accounted for 57.4% (11,587 reviews) and positive reviews represented 42.6% (8,613 reviews), as summarized in Table 5. This distribution reflects a realistic user behavior pattern, as users who encounter problems tend to be more motivated to provide feedback than those with neutral or positive experiences.

Table 5. Dataset Distribution

Sentiment Class	Number of Reviews	Percentage
Negative	11,587	57.4%
Positive	8,613	42.6%
Total	20,200	100%

For model development and evaluation, the dataset was divided into training and testing sets using a 70:30 ratio. Stratified sampling was applied to preserve the original class distribution in both subsets. The training set (n = 14,140) was utilized for model training and hyperparameter optimization through cross-validation, while the testing set (n = 6,060) was reserved exclusively for final performance evaluation.

Table 6. Best hyperparameter configurations and stratified 5-fold cross-validation performance

Kernel SVM	N-gram (TF-IDF)	Parameter SVM					Evaluation Metrics			
		C	*Max Iter*	*Gamma*	*Degree*	*Coef0*	*Accuracy*	*Precision*	*Recall*	*F1-Score*
Linear	(1, 1)	0.1	1000	-	-	-	0.8710	0.8712	0.8710	0.8706
Linear	(1, 2)	1	1500	-	-	-	0.8761	0.8792	0.8761	0.8750
Linear	(1, 3)	5	2000	-	-	-	0.8766	0.8789	0.8766	0.8757
RBF	(1, 1)	100	-	1	-	-	0.8746	0.8765	0.8746	0.8738
RBF	(1, 2)	50	-	0.01	-	-	0.8760	0.8795	0.8759	0.8748
RBF	(1, 3)	1000	-	0.1	-	-	0.8769	0.8798	0.8769	0.8759
Polynomial	(1, 1)	10	-	4	10	10	0.8741	0.8762	0.8741	0.8732
Polynomial	(1, 2)	5	-	2	0.01	10	0.8760	0.8795	0.8759	0.8748
Polynomial	(1, 3)	5	-	1	5	10	**0.8771**	**0.8803**	**0.8771**	**0.8760**
Sigmoid	(1, 1)	1	-	2	-	-1	0.8739	0.8755	0.8739	0.8732
Sigmoid	(1, 2)	10	-	0.1	-	0.1	0.8761	0.8796	0.8761	0.8749
Sigmoid	(1, 3)	500	-	0.1	-	-1	0.8764	0.8792	0.8764	0.8754

3.2. Hyperparameter Optimization Results

A comprehensive Grid Search with stratified 5-fold cross-validation was conducted on the training set to jointly optimize the parameters of the TF-IDF feature extractor and the SVM classifier. The search space for each kernel type was defined as follows:

- Linear kernel: C [0.001, 0.01, 0.1, 1, 5, 10, 50, 100, 500, 1000] and maximum iterations [10, 100, 500, 1000, 1500, 2000, 3500, 5000, 10000, -1].
- RBF kernel: C [0.001, 0.01, 0.1, 1, 5, 10, 50, 100, 500, 1000] and gamma [0.0001, 0.001, 0.01, 0.1, 0.5, 1, 2, 5, 10, 20].
- Polynomial kernel: C [0.001, 0.01, 0.1, 1, 5, 10, 50, 100, 500, 1000], gamma [0.0001, 0.001, 0.01, 0.1, 0.5, 1, 2, 5, 10, 20], degree [2, 3, 4, 5, 6, 7, 8, 9, 10, 12], and coefficient coef0 [-10, -5, -1, -0.1, 0, 0.1, 1, 5, 10, 100].
- Sigmoid kernel: C [0.001, 0.01, 0.1, 1, 5, 10, 50, 100, 500, 1000], gamma [0.0001, 0.001, 0.01, 0.1, 0.5, 1, 2, 5, 10, 20], and coef0 [-10, -5, -1, -0.1, 0, 0.1, 1, 5, 10, 100].

For each kernel and TF-IDF n-gram range combination [(1,1), (1,2), (1,3)], the hyperparameter set yielding the highest macro F1-score on the validation folds was selected. Table 6 presents these best configurations along with their corresponding cross-validation performance metrics, which include accuracy, macro-averaged precision, recall, and F1-score, each reported as the mean over the 5 folds.

The results in Table 6 demonstrate the impact of

hyperparameter tuning. The Polynomial kernel with trigram features (1,3) achieved the highest cross-validation macro F1-score (0.8760). Figure 2 visualizes the comparison of these F1-scores across kernels, showing that the Polynomial kernel with trigrams achieves the highest score, followed closely by the RBF and Linear kernels with similar n-gram ranges. The narrow performance margin (ΔF1 < 0.006) among the top configurations indicates that multiple kernel and parameter combinations yield comparable results for this classification task.

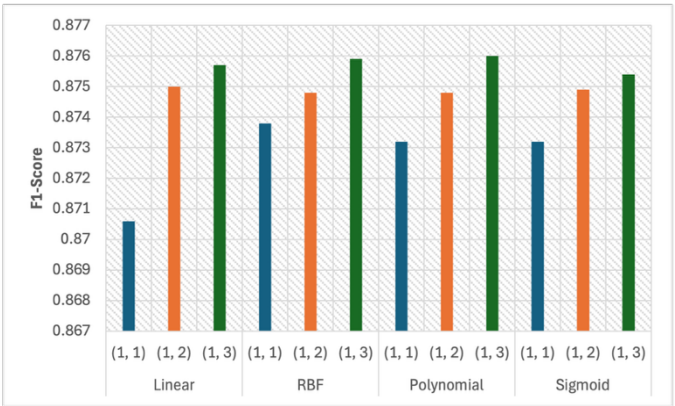

Figure 2. Macro F1-score across SVM kernels and n-gram ranges

3.3. Performance of the Optimized Model

The model with the optimal hyperparameters (Polynomial kernel, n-gram (1,3), C=5, gamma=1, degree=5, coef0=10) was retrained on the entire training set (14,140 reviews) and evaluated on the independent test set (6,060 reviews). The confusion matrix and detailed performance metrics are presented in Figure 3 and Table 7, respectively.

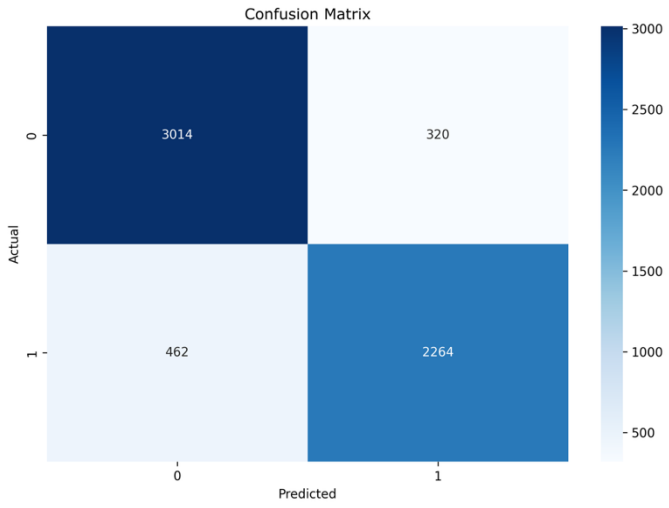

Figure 3. Confusion matrix of the optimized model

Table 7. Classification report for the optimized model

Metric	Negative Class	Positive Class	Macro Average	Weightes Average	Accuracy
Precision	0.9039	0.8305	0.8672	0.8723	–
Recall	0.8668	0.8762	0.8715	0.8706	–
F1-score	0.8850	0.8528	0.8689	0.8714	–
Support	3,477	2,583	6,060	6,060	–
Overall	–	–	–	–	0.8710

The optimized model achieved an overall accuracy of 87.1% and a macro F1-score of 86.9% on the independent test set. The performance is consistent with the cross-validation estimate (87.7% accuracy, 87.6% F1-score), indicating good generalization. The model exhibits high precision for the negative class (90.4%), making it particularly reliable for identifying critical user complaints, while maintaining balanced recall for both classes (86.7% negative, 87.6% positive).

4. Discussion

4.1. Interpretation of Optimal Hyperparameters

The Grid Search results presented in Table 9 reveal several critical insights into the sentiment classification task for Traveloka app reviews. The Polynomial kernel with trigram features (1,3) emerged as the optimal configuration, achieving a cross-validation macro F1-score of 0.8760 and a final test accuracy of 87.10%. The selection of a Polynomial kernel (characterized by its ability to model non-linear, higher-order feature interactions) suggests that the relationship between textual features and sentiment in this domain is not merely additive or linear. Instead, sentiment often emerges from complex interplay between words, where combinations and their multiplicative interactions (captured by the polynomial degree) carry significant meaning.

The importance of trigrams is further substantiated by the model's performance. While expanding the n-gram range from unigrams to bigrams yielded noticeable improvements (e.g., +0.44% F1-score for Linear kernel), the additional gain from bigrams to trigrams was more modest (+0.07% to +0.12% across kernels). This indicates that bigrams capture the majority of meaningful, sentiment-bearing phrases (e.g., "sangat puas", "tidak bisa"), while trigrams provide incremental value by capturing slightly more specific contexts (e.g., "pembatalan tiket mendadak"). The minimal performance gap among top configurations (ΔF1 < 0.006) indicates that the task is relatively robust to kernel and n-gram choices. Consequently, a Linear or RBF kernel with bigrams could serve as a more computationally efficient alternative with only a marginal sacrifice in accuracy for practical deployment scenarios.

The model's performance on the independent test set (F1-score 0.8689, accuracy 0.8710) aligns closely with the cross-validation estimate (F1-score 0.8760), demonstrating good generalization and suggesting that the Grid Search process did not lead to substantial overfitting despite the extensive hyperparameter search.

4.2. Qualitative Analysis of Linguistic Patterns

To gain insight into the linguistic characteristics of positive and negative reviews, word cloud visualizations were generated based on the most frequent terms in each class after preprocessing. Figure 4 (positive sentiment) and Figure 5 (negative sentiment) present these patterns.

Figure 5 (Positive) reveals terms associated with satisfaction and seamless experience, such as "baik" (good), "mudah" (easy), "cepat" (fast), "promo" (promotion), and "murah" (cheap). These reflect successful transactions, efficient service, and overall user

appreciation. In contrast, Figure 6 (Negative) is dominated by terms indicating problems and frustration, including "kecewa" (disappointed), "batal" (cancelled), "rugi" (loss), "buruk" (bad), "susah" (difficult), and "tidak" (not). The frequent appearance of "tapi" (but) also suggests that negative reviews often contain contrastive statements or mitigated complaints.

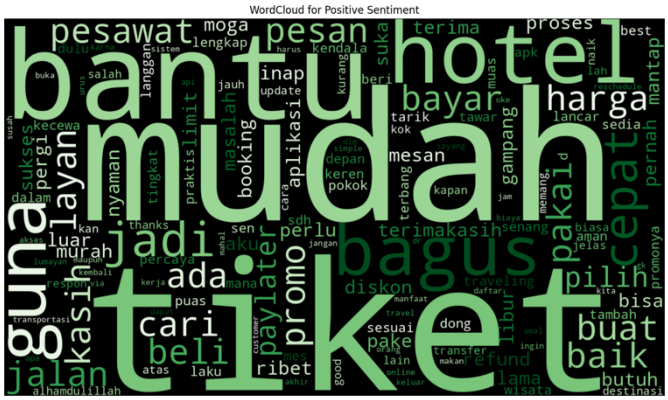

Figure 4. Positive sentiments in the Traveloka App

Figure 5. Negative sentiments in the Traveloka App

This clear lexical separation between sentiment classes helps explain the model's overall effectiveness. However, the presence of some overlapping neutral terms (e.g., "pesawat" – airplane, "hotel") in both clouds underscores why n-gram features are crucial: sentiment in reviews is often expressed through phrases and context (e.g., "pesawat murah" vs. *pesawat delay") rather than isolated keywords. These visual patterns provide qualitative support for the quantitative finding that expanding the n-gram range improves classification accuracy.

4.3. Error Analysis and Linguistic Challenges

Despite strong overall performance, the model exhibits an asymmetry in precision and recall between the two sentiment classes, as detailed in Table 11. The negative class achieves notably higher precision (0.9039) compared to the positive class (0.8305). This indicates that when the model predicts a review as negative, it is correct approximately 90% of the time, making it highly reliable for flagging critical feedback. In contrast, the lower precision for the positive class implies that a portion of reviews predicted as positive are actually negative (i.e., false positives). This could lead to missed critical complaints if deployed in an automated monitoring system.

Conversely, recall is more balanced (0.8668 for negative, 0.8762 for positive), indicating the model successfully identifies a similar proportion of actual instances from each class. This balance is commendable given the inherent class imbalance (57.4% negative), suggesting that stratified sampling and the use of macro-averaged metrics during optimization helped mitigate bias toward the majority class.

4.4. Limitations and Future Research Directions

This study acknowledges several limitations that inform potential future research. The reliance on a bag-of-ngrams representation (TF-IDF) inherently disregards word order, syntactic structure, and long-range contextual dependencies, which are crucial for understanding complex sentiments. While manual validation was applied, the initial heuristic labeling based on star ratings may introduce label noise, as textual sentiment does not always perfectly correlate with a numerical score. Furthermore, the model's scope is limited to binary classification; extending it to aspect-based sentiment analysis (ABSA) could provide more granular insights into user opinions about specific features like "customer service" or "payment process." The model's generalizability is also untested across different domains (e.g., other Indonesian e-commerce platforms), suggesting cross-domain adaptation as a valuable future direction. Finally, given the minimal performance differences between complex and simpler kernels, future work could involve a more comprehensive cost-benefit analysis, factoring in training efficiency, interpretability, and deployment latency, to determine the most practical model for real-world implementation.

5. Conclusion

This research systematically addresses the optimization of a sentiment analysis pipeline for Traveloka customer reviews through a comprehensive Grid Search of TF-IDF and SVM hyperparameters. The study successfully identified the Polynomial kernel with trigram features (C=5, gamma=1, degree=5, coef0=10) as the optimal configuration, achieving 87.10% accuracy and 86.9% macro F1-score on an independent test set. The close performance of alternative configurations, particularly the RBF kernel, indicates that the sentiment classification task for this Indonesian review dataset is relatively robust to specific hyperparameter choices, though systematic optimization yields measurable improvements over baseline approaches.

Key contributions of this work include: (1) a detailed empirical demonstration of how joint optimization of feature extraction (TF-IDF n-grams) and classifier (SVM kernel and parameters) significantly enhances model performance; (2) the establishment of a reproducible, optimized baseline using traditional machine learning methods for Indonesian sentiment analysis; and (3) valuable insights into the linguistic characteristics of Traveloka reviews, where multi-word expressions prove essential for accurate sentiment detection. The model's high precision for negative reviews (90.4%) presents a practical opportunity for deploying automated alert systems to prioritize critical customer feedback.

Despite these results, the study acknowledges the inherent limitations of the bag-of-ngrams approach, particularly its difficulty with complex linguistic structures such as contrastive clauses and contextual nuances. Future research should, therefore, focus on methodological advancements including the adoption of pre-trained contextual embeddings (e.g., IndoBERT) and the exploration of alternative architectures like neural networks or ensemble methods to capture deeper semantic meaning. Extending the analysis to aspect-based sentiment classification would also provide more granular and actionable business intelligence. This work thus provides both a validated framework for traditional model optimization and clear pathways for

methodological evolution in the domain of Indonesian language sentiment analysis.

Reference

[1] T. Pencarelli, "The digital revolution in the travel and tourism industry," *Information Technology & Tourism 2019*, vol. 22, no. 3, pp. 455–476, Nov. 2019, doi: 10.1007/S40558-019-00160-3.

[2] "Asosiasi Penyelenggara Jasa Internet Indonesia - Survei." Accessed: Jan. 23, 2026. [Online]. Available: https://survei1.apjii.or.id/

[3] N. Stylos, R. Rahimi, B. Okumus, and S. Williams, "Generation Z Marketing and Management in Tourism and Hospitality: The Future of the Industry," *Generation Z Marketing and Management in Tourism and Hospitality: The Future of the Industry*, pp. 1–332, May 2021, doi: 10.1007/978-3-030-70695-1.

[4] S. Li, F. Liu, Y. Zhang, B. Zhu, H. Zhu, and Z. Yu, "Text Mining of User-Generated Content (UGC) for Business Applications in E-Commerce: A Systematic Review," *Journal of Mathematics*, vol. 10, no. 19, Sep. 2022, doi: 10.3390/MATH10193554.

[5] V. Oktaviani, B. Warsito, H. Yasin, R. Santoso, and Suparti, "Sentiment analysis of e-commerce application in Traveloka data review on Google Play site using Naïve Bayes classifier and association method," *J. Phys. Conf. Ser.*, vol. 1943, no. 1, p. 012147, Jul. 2021, doi: 10.1088/1742-6596/1943/1/012147.

[6] Y. Mao, Q. Liu, and Y. Zhang, "Sentiment analysis methods, applications, and challenges: A systematic literature review," *Journal of King Saud University - Computer and Information Sciences*, vol. 36, no. 4, p. 102048, Apr. 2024, doi: 10.1016/J.JKSUCI.2024.102048.

[7] N. M. Gardazi, A. Daud, M. K. Malik, A. Bukhari, T. Alsahfi, and B. Alshemaimri, "BERT applications in natural language processing: a review," *Artificial Intelligence Review 2025 58:6*, vol. 58, no. 6, pp. 166-, Mar. 2025, doi: 10.1007/S10462-025-11162-5.

[8] W. Ansar, A. K. Choudhury, S. Goswami, and A. Chakrabarti, "From Transformers to LLMs: A Systematic Survey of Efficiency Considerations in NLP," *Jpurnal of arXiv*, May 2024, Accessed: Jan. 23, 2026. [Online]. Available: https://arxiv.org/pdf/2406.16893

[9] D. M. Abdullah and A. M. Abdulazeez, "Machine Learning Applications based on SVM Classification A Review," *Qubahan Academic Journal*, vol. 1, no. 2, pp. 81–90, Apr. 2021, doi: 10.48161/QAJ.V1N2A50.

[10] J. Shin *et al.*, "Exploring the Effectiveness of Machine Learning and Deep Learning Algorithms for Sentiment Analysis: A Systematic Literature Review," *Computers, Materials & Continua*, vol. 84, no. 3, pp. 4105–4153, Jul. 2025, doi: 10.32604/CMC.2025.066910.

[11] R. I. Alhaqq, I. Made, K. Putra, and Y. Ruldeviyani, "Analisis Sentimen terhadap Penggunaan Aplikasi MySAPK BKN di Google Play Store," *Jurnal Nasional Teknik Elektro dan Teknologi Informasi* |, vol. 11, no. 2, 2022.

[12] Kusnawi, M. Rahardi, and V. D. Pandiangan, "Sentiment Analysis of Neobank Digital Banking using Support Vector Machine Algorithm in Indonesia," *JOIV : International Journal on Informatics Visualization*, vol. 7, no. 2, pp. 377–383, May 2023, doi: 10.30630/JOIV.7.2.1652.

[13] M. Isnan, G. N. Elwirehardja, and B. Pardamean, "Sentiment Analysis for TikTok Review Using VADER Sentiment and SVM Model," *Procedia Comput. Sci.*, vol. 227, pp. 168–175, Jan. 2023, doi: 10.1016/J.PROCS.2023.10.514.

[14] S. Fransiska, R. Rianto, and A. I. Gufroni, "Sentiment Analysis Provider By.U on Google Play Store Reviews with TF-IDF and Support Vector Machine (SVM) Method," *Scientific Journal of Informatics*, vol. 7, no. 2, pp. 203–212, Nov. 2020, doi: 10.15294/SJI.V7I2.25596.

[15] K. Alemerien, A. Al-Ghareeb, and M. Z. Alksasbeh, "Sentiment Analysis of Online Reviews: A Machine Learning Based Approach with TF-IDF Vectorization," *Journal of Mobile Multimedia*, vol. 20, no. 5, pp. 1089–1116, 2024, doi: 10.13052/JMM1550-4646.2055.

[16] R. H. Muhammadi, T. G. Laksana, and A. B. Arifa, "Combination of Support Vector Machine and Lexicon-Based Algorithm in Twitter Sentiment Analysis," *Khazanah Informatika : Jurnal Ilmu Komputer dan Informatika*, vol. 8, no. 1, pp. 59–71, Mar. 2022, doi: 10.23917/KHIF.V8I1.15213.

[17] C. A. Nurhaliza Agustina, R. Novita, Mustakim, and N. E. Rozanda, "The Implementation of TF-IDF and Word2Vec on Booster Vaccine Sentiment Analysis Using Support Vector Machine Algorithm," *Procedia Comput. Sci.*, vol. 234, pp. 156–163, Jan. 2024, doi: 10.1016/J.PROCS.2024.02.162.

[18] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, Nov. 2020, doi: 10.1016/J.NEUCOM.2020.07.061.

[19] N. Subaşı, "European Journal of Engineering and Applied Sciences Comprehensive Analysis of Grid and Randomized Search on Dataset Performance," *App. Sci*, vol. 7, no. 2, pp. 77–83, 2024, doi: 10.55581/ejeas.1581494.

[20] J. Bergstra, J. B. Ca, and Y. B. Ca, "Random Search for Hyper-Parameter Optimization Yoshua Bengio," *Journal of Machine Learning Research*, vol. 13, pp. 281–305, 2012, Accessed: Jan. 23, 2026. [Online]. Available: http://scikit-learn.sourceforge.net.

[21] V. W. Lumumba, D. Kiprotich, M. L. Mpaine, N. G. Makena, and M. D. Kavita, "Comparative Analysis of Cross-Validation Techniques: LOOCV, K-folds Cross-Validation, and Repeated K-folds Cross-Validation in Machine Learning Models," *American Journal of Theoretical and Applied Statistics 2024, Volume 13, Page 127*, vol. 13, no. 5, pp. 127–137, Oct. 2024, doi: 10.11648/J.AJTAS.20241305.13.

[22] A. Q. Md, S. Kulkarni, C. J. Joshua, T. Vaichole, S. Mohan, and C. Iwendi, "Enhanced Preprocessing Approach Using Ensemble Machine Learning Algorithms for Detecting Liver Disease," *Biomedicines 2023, Vol. 11,* vol. 11, no. 2, Feb. 2023, doi: 10.3390/BIOMEDICINES11020581.

[23] A. N. Ma'aly, D. Pramesti, A. D. Fathurahman, and H. Fakhrurroja, "Exploring Sentiment Analysis for the Indonesian Presidential Election Through Online Reviews Using Multi-Label Classification with a Deep Learning Algorithm," *Information 2024, Vol. 15,* vol. 15, no. 11, Nov. 2024, doi: 10.3390/INFO15110705.

[24] C. J. Harrison and C. J. Sidey-Gibbons, "Machine learning innmedicine: aapractical introduction toonatural language processing," *BMC Med Res Methodol*, vol. 21, p. 158, 2021, doi: 10.1186/s12874-021-01347-1.

[25] K. Park, J. Lee, S. Jang, D. Jung, and K. Brain, "An Empirical Study of Tokenization Strategies for Various Korean NLP Tasks," pp. 133–142, Accessed: Jan. 23, 2026. [Online]. Available: http://www.aihub.or.kr/aidata/87

[26] D. J. Ladani and N. P. Desai, "Stopword Identification and Removal Techniques on TC and IR applications: A Survey," *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, pp. 466–472, Mar. 2020, doi: 10.1109/ICACCS48705.2020.9074166.

[27] A. Jabbar, S. Iqbal, M. I. Tamimy, S. Hussain, and A. Akhunzada, "Empirical evaluation and study of text stemming algorithms," *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5559–5588, Dec. 2020, doi: 10.1007/S10462-020-09828-3.

[28] A. Onan and M. A. Tocoglu, "A Term Weighted Neural Language Model and Stacked Bidirectional LSTM Based Framework for Sarcasm Identification," *IEEE Access*, vol. 9, pp. 7701–7722, 2021, doi: 10.1109/ACCESS.2021.3049734.

[29] N. S. Mohd Nafis and S. Awang, "An Enhanced Hybrid Feature Selection Technique Using Term Frequency-Inverse Document Frequency and Support Vector Machine-Recursive Feature Elimination for Sentiment Classification," *IEEE Access*, vol. 9, pp. 52177–52192, 2021, doi: 10.1109/ACCESS.2021.3069001.

[30] C. Cortes, V. Vapnik, and L. Saitta, *Support-vector networks*, vol. 20, no. 3. Springer, 1995. doi: 10.1007/BF00994018.

[31] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine

classification: Applications, challenges and trends," *Neurocomputing*, vol. 408, pp. 189–215, Sep. 2020, doi: 10.1016/J.NEUCOM.2019.10.118.

[32] L. Wang and X. Shen, "On L1-Norm Multiclass Support Vector Machines," *J. Am. Stat. Assoc.*, vol. 102, no. 478, pp. 583–594, Jun. 2007, doi: 10.1198/016214506000001383.

[33] Gustavo. Camps-Valls and Lorenzo. Bruzzone, *Kernel methods for remote sensing data analysis*. Wiley, 2009.

[34] P. Tao, Z. Sun, and Z. Sun, "An Improved Intrusion Detection Algorithm Based on GA and SVM," *IEEE Access*, vol. 6, pp. 13624–13631, Mar. 2018, doi: 10.1109/ACCESS.2018.2810198.

[35] R. Rodríguez-Pérez and J. Bajorath, "Evolution of Support Vector Machine and Regression Modeling in Chemoinformatics and Drug Discovery," *Journal of Computer-Aided Molecular Design 2022 36:5*, vol. 36, no. 5, pp. 355–362, Mar. 2022, doi: 10.1007/S10822-022-00442-9.

[36] I. Malashin, V. Tynchenko, A. Gantimurov, V. Nelyub, and A. Borodulin, "Support Vector Machines in Polymer Science: A Review," *Polymers 2025, Vol. 17,* vol. 17, no. 4, Feb. 2025, doi: 10.3390/POLYM17040491.

[37] G. Zeng, "Invariance Properties and Evaluation Metrics Derived from the Confusion Matrix in Multiclass Classification," *Mathematics 2025, Vol. 13,* vol. 13, no. 16, Aug. 2025, doi: 10.3390/MATH13162609.

[38] A. N. Jahromi, S. Hashemi, A. Dehghantanha, R. M. Parizi, and K. K. R. Choo, "An Enhanced Stacked LSTM Method With No Random Initialization for Malware Threat Hunting in Safety and Time-Critical Systems," *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 4, no. 5, pp. 630–640, Oct. 2020, doi: 10.1109/TETCI.2019.2910243.