BAREKENG: Journal of Mathematics and Its Applications
September 2026 Volume 20 Issue 3 Page 2575-2586
P-ISSN: 1978-7227 E-ISSN: 2615-3017
https://doi.org/10.30598/barekengvol20iss3pp2575-2586

ENHANCING E-COMMERCE REVIEW SENTIMENT ANALYSIS WITH LINEAR SVM: FEATURE-EXTRACTION AND HYPERPARAMETER COMPARISONS

Fauziah Hanum[1*], **Richi Andrianto**[2], **Anita Sri Rejeki Hutagaol**[3], **Nurhanna Harahap**[4], **Ibnu Rasyid Munthe**[5]

[1,3] Faculty of Economics and Business, Universitas Labuhanbatu
[5] Faculty of Science and Technology, Universitas Labuhanbatu
Jln Sisingamangaraja, Labuhanbatu, 21418, Indonesia

[2] Information Systems, Institut Teknologi dan Sains Padang Lawas Utara
Jln. Lintas Gunungtua, Padang Sidempuan, Padang Lawas Utara, 22740, Indonesia

[4] Faculty of Teacher Training and Education, Universitas Al Washliyah Labuhanbatu
Jln. H. Adam Malik Lingkar Bypass, Labuhanbatu, 21418, Indonesia

*Corresponding author's e-mail: * **fauziahhanummrp@gmail.com***

Article Info

Article History:

Received: 15th September 2025
Revised: 21st January 2026
Accepted: 17th March 2026
Published: 8th April 2026

Keywords:

E-commerce;
Feature Extraction;
Hyperparameter tuning;
Sentiment Analysis;
SVM Linear.

ABSTRACT

Sentiment analysis of e-commerce reviews is essential for understanding customer perceptions and supporting service and marketing decisions. However, previous SVM-based studies often report results using only one feature representation or one tuning approach, which provides limited guidance on the most effective practical configuration. This study addresses this gap by benchmarking a linear Support Vector Machine across TF IDF and Word2Vec representations and comparing three hyperparameter tuning strategies, Grid Search, Random Search, and Optuna, on an Indonesian language dataset of customer product reviews. The held-out test set contains 871 reviews, while class imbalance in the training data is handled by applying SMOTE only on the training set, resulting in a balanced training set of 2902 samples. Using stratified validation with Accuracy, Precision, Recall, F1 score, and ROC AUC, the best configuration is TF IDF with Optuna-tuned linear SVM, achieving 86.68 percent accuracy, an F1 score of 0.87, and ROC AUC of about 0.93 to 0.94. For Word2Vec, the best result is obtained with Random Search, reaching 84.38 percent accuracy, an F1 score of 0.84, and an ROC AUC of about 0.92. These findings indicate that TF-IDF is a stronger match for linear SVM in this setting, and that Optuna provides the most consistent gains for TF-IDF. Limitations include the use of binary sentiment labels and an evaluation focused on linear SVM with simple document-level Word2Vec aggregation, so performance may differ across other domains, platforms, and languages. Future research will examine richer document embeddings, nonlinear and contextual models, multi-class or aspect-level sentiment, and broader cross-platform validation to improve generalizability.

How to cite this article:

F. Hanum, R. Andrianto, A. S. R. Hutagaol, N. Harahap and I. R. Munthe., "ENHANCING E-COMMERCE REVIEW SENTIMENT ANALYSIS WITH LINEAR SVM: FEATURE-EXTRACTION AND HYPERPARAMETER COMPARISONS", *BAREKENG: J. Math. & App.*, vol. 20, no. 3, pp. 2575-2586, Sep, 2026.

1. INTRODUCTION

The rapid growth of e-commerce has generated massive amounts of data, particularly customer reviews that capture opinions, perceptions, and satisfaction levels regarding products and services [1]. This information is strategically valuable for companies because it can be used to understand customer needs, improve service quality, and design effective marketing strategies [2]. However, customer reviews are unstructured text, so natural language processing (NLP) methods are required to process them for systematic analysis [3]. One of the most widely used machine learning algorithms for sentiment analysis is the Support Vector Machine (SVM). SVM is known to perform well on high-dimensional data, including text, and to deliver stable classification performance [4]. The algorithm constructs an optimal hyperplane that maximizes the margin between classes [5]. Nevertheless, SVM performance strongly depends on the quality of preprocessing, the feature-extraction technique, and the choice of hyperparameters [6].

Prior studies underscore SVM's effectiveness for sentiment classification tasks. Study [7] reported that SVM with TF-IDF achieves high accuracy for classifying Tokopedia product reviews. [8] compared SVM and Naïve Bayes on Shopee reviews and found SVM to be more consistent. Study [9] applied Word2Vec with SVM on Amazon reviews and improved text representation quality compared to simple bag-of-words approaches. Study [10] combined SVM with Information Gain feature selection for Twitter sentiment analysis and obtained an accuracy above 89%. Meanwhile, [11] demonstrated that hyperparameter tuning using Grid Search improves SVM performance on the IMDb Movie Reviews dataset. Building on these findings, SVM remains competitive for sentiment analysis, yet performance can still be improved along two main axes: feature representation and hyperparameter optimization [12]. owever, although the literature has demonstrated the benefits of TF-IDF, Word2Vec, and hyperparameter tuning in various contexts, many studies still report using a single feature representation or a single tuning strategy, which makes it difficult to determine, under a consistent evaluation protocol, which combination is the most reliable and efficient for practical e-commerce sentiment analysis [12]. In particular, a controlled comparison that jointly evaluates TF-IDF versus Word2Vec and simultaneously contrasts Grid Search, Random Search, and Optuna within the same SVM pipeline remains limited, even though each component has been shown in prior work to affect performance.

First, this study compares two feature-extraction methods: TF-IDF and Word2Vec. TF-IDF is selected for its simplicity and its ability to weight terms by importance with relatively low computational cost [13]. In addition, TF-IDF typically yields sparse, high-dimensional feature vectors, a setting where linear SVMs are commonly adopted because they are effective and computationally efficient for large-scale text classification [14]. By contrast, Word2Vec captures semantic meaning and word relationships, producing dense vector representations that better reflect contextual similarity in language use [15]. However, when Word2Vec is converted into document-level features via simple averaging, the representation may lose word-order information and weaken compositional meaning modeling; therefore, Word2Vec's effectiveness can depend on how document embeddings are constructed and on domain characteristics [16]. In this context, the present study positions TF-IDF and Word2Vec as two representatives yet contrasting feature paradigms and systematically evaluates them in the same experimental setting to clarify which representation is more suitable for SVM-based e-commerce review sentiment classification.

Second, this study emphasizes hyperparameter optimization for SVMs. The commonly used Grid Search evaluates all parameter combinations within a predefined grid but can be computationally expensive [17]. As a comparison, Random Search can be more efficient because it samples parameter configurations randomly and often finds strong solutions with fewer evaluations, particularly when only a subset of hyperparameters substantially affects performance [18]. In addition, we use Optuna, a modern optimization framework that supports Bayesian optimization and pruning, enabling more adaptive, computationally efficient search by allocating fewer resources to unpromising trials [19]. Accordingly, this research focuses on developing an e-commerce sentiment analysis model using SVM, comparing two feature-extraction techniques (TF-IDF and Word2Vec) and three hyperparameter optimization methods (Grid Search, Random Search, and Optuna), with the objective of identifying the most effective and practically efficient configuration within a common evaluation pipeline. The evaluation uses Accuracy, Precision, Recall, F1-score, and ROC-AUC to comprehensively assess model quality, and the results are expected to contribute to theoretical understanding of machine-learning–based sentiment modeling while providing practical benefits for e-commerce stakeholders by enabling a more accurate understanding of consumer perceptions.

2. RESEARCH METHODS

The rapid growth of online shopping has made e-commerce reviews a valuable source of insights for both customers and businesses. However, the sheer volume of textual reviews requires automated sentiment analysis techniques that are both efficient and accurate. In this context, Support Vector Machine (SVM) remains one of the most reliable classifiers for text categorization, particularly when enhanced with appropriate preprocessing, feature extraction, and hyperparameter optimization.

This article presents a methodological approach to improve the accuracy of e-commerce review sentiment analysis using a linear kernel SVM. The preprocessing consists of case folding, tokenization, stopword removal, and stemming. Two feature-extraction schemes are compared: TF-IDF, which is sparse and efficient, and Word2Vec, which is dense and semantic, to assess how representation affects SVM performance. Three hyperparameter-tuning strategies, Grid Search, Random Search, and Optuna, are evaluated separately for each feature set to obtain the best configuration. Model quality is measured using Accuracy, Precision, Recall, F1-score, and ROC-AUC under stratified validation. Fig. 1 details the methodological flow from data collection to evaluation.

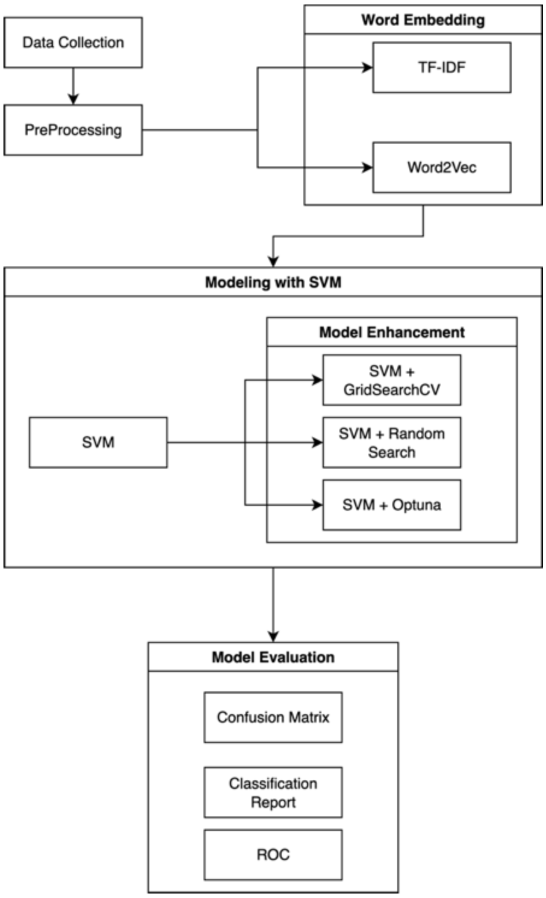

Figure 1. Methodology Flow

2.1 Dataset

The data consist of customer reviews from e-commerce platforms containing opinion text and sentiment labels (positive/negative). Sources may be public datasets or controlled scraping compliant with ethics and terms of service. Each entry includes at least a review (text) and a label (class), with initial cleaning such as duplicate removal, unified encoding (UTF-8), and handling missing values (dropping rows with null/empty reviews). To ensure a valid evaluation, the data are split into training and test sets using a stratified split (e.g., 80:20) to preserve class proportions. To address the imbalance that can bias the classifier, this study applies SMOTE [20], [21], [22]. The balanced distribution after SMOTE is shown below.

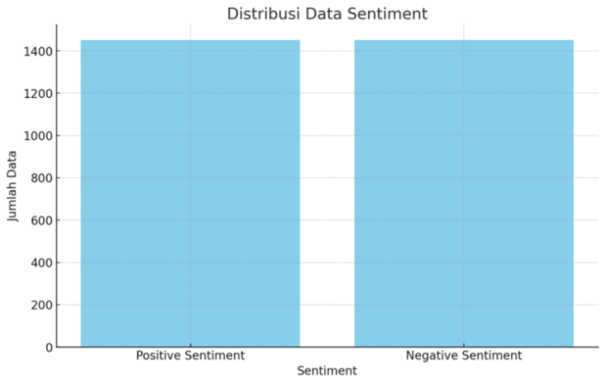

Figure 2. Data Balancing Using SMOTE

Fig. 2 shows that after applying SMOTE, the two classes are perfectly balanced: 1 = 1,451 samples and 0 = 1,451 samples. SMOTE augments only the original minority class by synthesizing new examples through interpolation in feature space (e.g., TF-IDF or Word2Vec) between a minority sample and its nearest neighbor, so the added data are not duplicates. This balancing is important for a linear SVM because it reduces hyperplane bias toward the majority class; typical effects include higher recall on the minority class and improved macro-F1, though precision may decrease slightly as a trade-off [23]. For validity, SMOTE must be applied only to the training data (not the test set) within a pipeline alongside preprocessing and feature extraction to prevent data leakage, and it should not be combined with class_weight='balanced' to avoid double compensation.

2.2 Experimental Environment

The experiments were conducted on a personal computer running Windows 11. The system is equipped with an Intel Core i7-class processor and 16 GB of RAM, and no dedicated GPU was required because all models were trained on the CPU. The implementation was developed in Python version 3.10. Key libraries used in this study include scikit-learn version 1.3, imbalanced learn version 0.11 for SMOTE, gensim version 4.3 for Word2Vec, Optuna version 3.5 for hyperparameter optimization, NumPy version 1.24, pandas version 2.0, and Matplotlib version 3.7 for visualization. To ensure reproducibility, a fixed random seed was applied where applicable, and all experiments followed the same stratified train-test split and evaluation protocol.

2.3 Preprocessing

This stage prepares text to be consistent and informative for modeling [24], [25]. Steps include case folding (lowercasing), tokenization (splitting text into tokens/words), stopword removal (removing common non-informative words such as “and,” “which,” “in”), and stemming (reducing words to their root forms with an Indonesian stemmer). Lemmatization is not used; consistency is achieved through stemming. Additional cleaning removes irrelevant non-alphabetic characters (e.g., certain emojis/symbols), normalizes whitespace, and strips URLs, mentions, and hashtags. The result is a clean, standardized corpus ready for feature extraction.

2.4 Feature Extraction

TF-IDF converts the corpus into a term–document matrix whose weights emphasize words that are important in a document yet rare across the corpus [26], [27], [28]. Configurations explored include ngram_range ((1,1) unigram or (1,2) uni+bigram), max_features in the range 10k–100k to control dimensionality, min_df/max_df to filter overly rare or overly common terms, and sublinear_tf=True to apply log-scaled term frequency. TF-IDF is chosen for its simplicity, speed, and effectiveness on high-dimensional text—often a strong baseline for SVM. Vectors can be L2-normalized to standardize feature scales, which typically benefits a linear SVM.

To capture semantic information and contextual proximity, Word2Vec maps each word to a fixed-dimensional dense vector [29], [30], [31]. A CBOW model is trained on the study corpus to adapt its embeddings to the e-commerce review domain; hyperparameters explored include vector_size, window, min_count, and epochs. Document representations are formed by mean pooling (averaging word vectors per review) or TF-IDF-weighted averaging, producing a single vector per review consumable by SVM. Thus, Word2Vec provides semantic signals complementary to TF-IDF’s frequency-based signals.

2.5 Modeling

The classifier is a Support Vector Machine (SVM), selected for its ability to handle high-dimensional data with a maximum-margin decision boundary [32]. Let $x_i \in \mathbb{R}^d$ denote the feature vector of i-th document (e.g., TF-IDF or an embedding-based representation), where d is the number of features, and let $y_i \in \{-1, +1\}$ denote its class label (negative and positive, respectively). A linear SVM defines a decision function $f(x) = w^T x + b$, where $w \in \mathbb{R}^d$ is the weight vector and $b \in \mathbb{R}$ is the bias term. The separating hyperplane is therefore given by:

$$\omega^T x + b = 0. \tag{1}$$

In practice, sentiment data are rarely perfectly separable; hence, we employ the soft-margin SVM, which introduces slack variables $\xi_i \geq 0$ to allow limited misclassification. The optimization problem is formulated as:

$$\min_{w,b,\xi} \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^{n} \xi_i, \tag{2}$$

subject to:

$$y_i(\omega^T \chi_i + b) \geq 1, \xi_i, i = 1,2, \dots, n, \tag{3}$$

where n is the number of training instances, ||w|| is the Euclidean norm of w, and C > 0 is the regularization hyperparameter controlling the trade-off between maximizing the margin and penalizing margin violations. For evaluation using ROC-AUC, SVM outputs are obtained from the decision score $f(x)$ via decision_function; optionally, probability=true can be enabled to provide calibrated probabilities through Platt scaling. Overall, the maximum-margin principle and effective regularization make SVM suitable for sentiment analysis, particularly when the feature representation is high-dimensional and sparse.

2.6 Hyperparameter Tuning

Three tuning approaches are compared:

1. Grid Search evaluates all combinations on a predefined grid:

$$\text{C} \in \{0.1,\ 1,\ 10,\ 100\}. \tag{4}$$

   It is exhaustive and transparent, but becomes computationally expensive as the grid grows.

2. Random Search samples parameter combinations at random, with C drawn from a log-uniform range

$$C \sim \text{loguniform}(10^{-2}, 10^{2}). \tag{5}$$

   For a limited number of trials. It often finds near-optimal settings at much lower cost and is well-suited to large spaces.

3. Optuna optimizes the objective adaptively using a Bayesian/TPE sampler and supports pruning/early stopping for unpromising trials. The optimization objective can be written as:

$$min_c \mathcal{L}(C), \tag{6}$$

   where the function represents the loss (or error) function evaluated under cross-validation. Optuna further enhances efficiency by supporting pruning and early stopping, enabling it to terminate unpromising trials early. This adaptive strategy enables Optuna to allocate more resources to promising regions of the search space, making it more efficient and often more effective than exhaustive or purely random search methods.

All three approaches are run separately for each feature scenario (TF-IDF and Word2Vec), yielding the top accuracy for each feature × tuner combination.

2.7 Evaluation and Analysis

Final evaluation is conducted on a held-out test set using Accuracy, Precision, Recall, F1-score (macro/micro), and ROC-AUC.

1. Accuracy

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \tag{7}$$

Measures the overall proportion of correctly classified instances out of the total number of instances.

2. Precision

$$Precision = \frac{TP}{TP + FP}. \tag{8}$$

Indicates the proportion of true positive predictions among all positive predictions made by the model, reflecting how reliable the positive classifications are.

3. Recall

$$Recall = \frac{TP}{TP + FN}. \tag{9}$$

Represents the proportion of true positives identified out of all actual positives, showing how well the model captures positive cases.

4. F1-score

$$F1 = 2X\frac{Precision\ x\ Recall}{Precision + \ Recall}. \tag{10}$$

Combines precision and recall into a single metric. Macro F1-score calculates the average across all classes equally, while Micro F1-score aggregates the contributions of all classes, making it more sensitive to class imbalance.

5. ROC-AUC

$$AUC = \int_0^1 TPR(FPR)\, d(FPR). \tag{11}$$

Represents the area under the Receiver Operating Characteristic curve, where the True Positive Rate (TPR) is plotted against the False Positive Rate (FPR). Higher values indicate better discrimination between positive and negative classes.

A classification report and confusion matrix are included to analyze class-wise errors. Results are compared along two dimensions: (a) feature representations (TF-IDF vs. Word2Vec) with SVM, and (b) tuning methods (Grid, Random, Optuna) within each feature setting.

3. RESULTS AND DISCUSSION

This section presents the experimental results and analysis of the linear-kernel SVM for e-commerce sentiment classification. We begin with a summary of the dataset and class rebalancing using SMOTE, then compare two feature-extraction schemes (TF-IDF and Word2Vec). Next, we evaluate three hyperparameter-tuning strategies, Grid Search, Random Search, and Optuna, under stratified validation. Performance is reported with Accuracy, Precision, Recall, F1-score, and ROC-AUC.

3.1 Experiments with TF-IDF

The initial experiment uses TF-IDF with Optuna; the result is shown in Fig. 3.

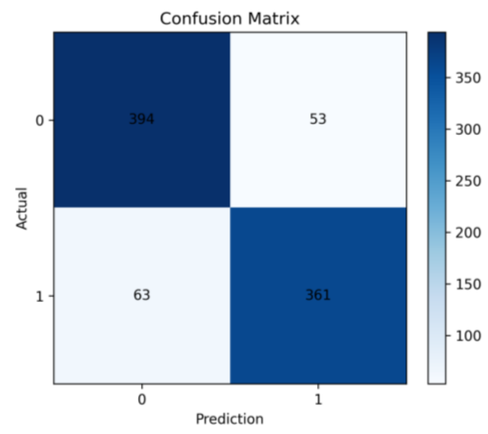

Figure 3. Confusion Matrix of SVM with TF-IDF and Optuna

Fig. 3 presents the confusion matrix of the linear SVM optimized by Optuna on the held-out test set of 871 instances (class 0 = 447, class 1 = 424). The model correctly classifies 394 instances of class 0 as class 0 (true negatives) and 361 instances of class 1 as class 1 (true positives). The remaining errors consist of 53 instances of class 0 predicted as class 1 (false positives) and 63 instances of class 1 predicted as class 0 (false negatives). Since false negatives exceed false positives, the classifier is slightly more conservative in predicting the positive class, so some positive reviews are still missed. The corresponding classification report is provided in Fig. 4.

```
              precision    recall  f1-score   support

           0       0.86      0.88      0.87       447
           1       0.87      0.85      0.86       424

    accuracy                           0.87       871
   macro avg       0.87      0.87      0.87       871
weighted avg       0.87      0.87      0.87       871

Accuracy : 0.8668197474167624
```

Figure 4. Classification Report of SVM with TF-IDF and Optuna

Fig. 4 reports results for 871 test instances (class 0 = 447, class 1 = 424). The model attains an accuracy of 0.8668 (approximately 86.7 percent) with balanced classwise performance. For class 0, precision is 0.86, recall is 0.88, and the F1 score is 0.87, indicating strong detection performance for this class. For class 1, precision is 0.87, recall is 0.85, and the F1 score is 0.86, indicating that positive predictions are relatively reliable while a portion of positives is still missed, consistent with the higher false-negative count in the confusion matrix. Macro and weighted averages are both 0.87, suggesting stable performance that is not driven by class proportions. The ROC analysis is shown in Fig. 5.

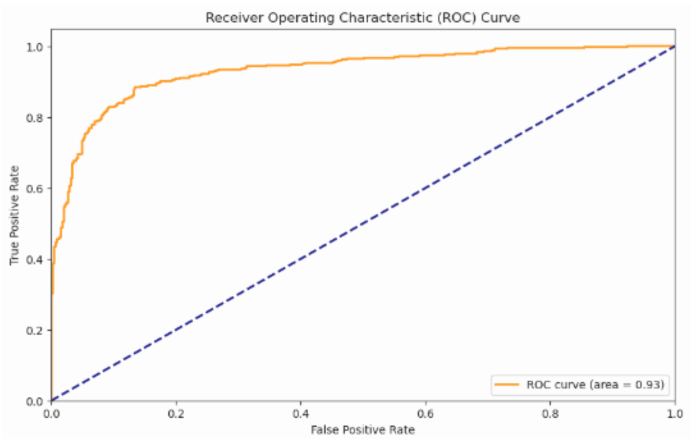

Figure 5. ROC of SVM with TF-IDF and Optuna

Fig. 5 shows the ROC curve of the best TF-IDF configuration, namely the Optuna-tuned linear SVM. The curve bends toward the upper-left corner and remains well above the diagonal line representing random performance, indicating strong class separability. The ROC-AUC is approximately 0.93, indicating the model is likely to assign a higher decision score to a randomly selected positive review than to a randomly selected negative review. At a false-positive rate of about 0.10, the true-positive rate reaches approximately 0.75 to

0.80, suggesting that many positive reviews can be detected while keeping false alarms relatively low. AUC values for all TF-IDF models are summarized in Table 1, while Fig. 5 is presented as a representative ROC visualization for the best-performing model.

Table 1. All TF-IDF Models

Models	Accuracy	Precision	Recall	F1 Score	ROC
SVM	84.38%	84%	84%	84%	93%
SVM + Random Search	85.87%	86%	86%	86%	93%
SVM + Grid Search	84.73%	85%	85%	85%	92%
SVM + Optuna	86.68%	87%	87%	87%	93%

Table 1 summarizes performance for all linear-SVM configurations with TF-IDF. The baseline SVM achieves 84.38% accuracy, F1 = 84%, and ROC ≈ 93%. Random Search lifts metrics to 85.87% accuracy and F1 = 86% (about +1.5 points over baseline) with the same ROC. Grid Search yields limited gains (84.73%, F1 = 85%) and a slight decrease in ROC (92%), indicating the grid did not cover the best region. Optuna yields the top result—86.68% accuracy, F1 = 87%, ROC 93%—about +2.3 accuracy points and +3 F1 points over baseline. Since AUC stays nearly constant (92–93%), global separability is already strong; improvements mainly come from better C and related settings that balance precision–recall. Thus, TF-IDF + linear SVM + Optuna is the preferred configuration.

3.2 Experiments with Word2vec

The next experiment uses Word2Vec with Random Search; the result is shown in Fig. 6.

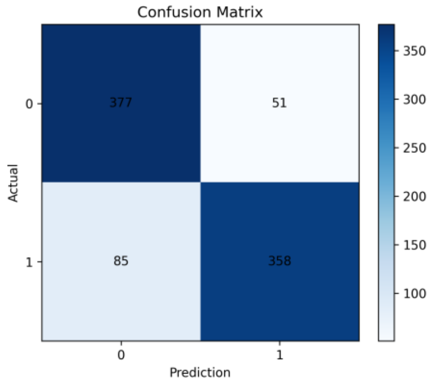

Figure 6. Confusion Matrix of SVM with Word2Vec and Random Search

Fig. 6 presents the confusion matrix for the linear SVM using Word2Vec features, optimized by Random Search, on the same held-out test set used in the TF-IDF experiments, consisting of 871 reviews (class 0 = 428, class 1 = 443). The model correctly classifies 377 instances of class 0 as class 0 (true negatives) and 358 instances of class 1 as class 1 (true positives). The remaining errors include 51 instances of class 0 misclassified as class 1 (false positives) and 85 instances of class 1 misclassified as class 0 (false negatives). Since the number of false negatives exceeds the number of false positives, the classifier exhibits a more conservative behavior when predicting the positive class, leading to higher precision but lower recall for positive sentiment. The corresponding classification report is shown in Fig. 7.

```
              precision    recall  f1-score   support

           0       0.82      0.88      0.85       428
           1       0.88      0.81      0.84       443

    accuracy                           0.84       871
   macro avg       0.85      0.84      0.84       871
weighted avg       0.85      0.84      0.84       871

Accuracy : 0.8438576349024111
```

Figure 7. Classification Report of SVM with Word2Vec and Random Search

On 871 test instances (class 0 = 428, class 1 = 443), the model achieves an accuracy ≈ 0.848. For class 0, precision 0.82, recall 0.88, F1 0.85, showing the model captures negatives well (few FN). For class 1, precision 0.88, recall 0.81, F1 0.84, predictions are cautious (low FP), but some positives are missed (higher FN), consistent with the confusion matrix. Macro and weighted F1 both equal 0.84, indicating stable performance across classes. The ROC curve is given in Fig. 8.

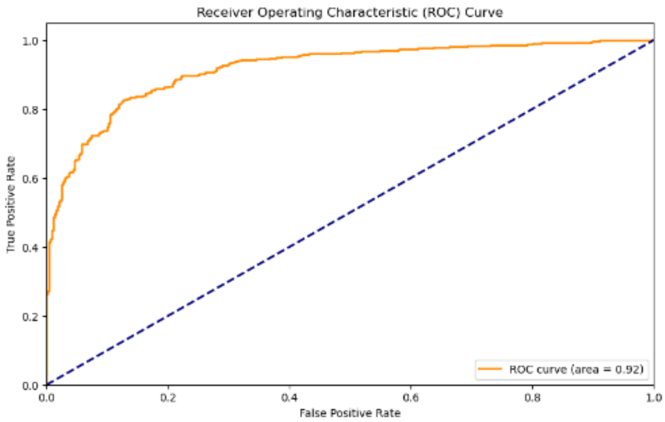

Figure 8. ROC of SVM with Word2Vec and Random Search

The ROC curve lies well above the diagonal, with AUC ≈ 0.91, confirming good separability, though slightly below that of the TF-IDF + Optuna configuration. At FPR ≈ 0.10 and TPR ≈ 0.75, many positives are detected while keeping false alarms modest. The threshold can be adjusted at the Youden's J maximum to raise positive recall if required. A full comparison of Word2Vec models is shown in Table 2.

Table 2. All Word2vec Models

Models	Accuracy	Precision	Recall	F1 Score	ROC
SVM	80.71%	81%	81%	81%	89%
SVM + Random Search	84.38%	85%	84%	84%	92%
SVM + Grid Search	83.58%	84%	84%	84%	92%
SVM + Optuna	83.81%	84%	84%	84%	91%

Table 2 shows that tuning improves performance by 3–4 points relative to the Word2Vec baseline. Random Search performs best (84.38% accuracy, F1 = 84%, AUC 92%), followed by Grid Search and Optuna. Although AUC improves from 0.89 to 0.91–0.92, Word2Vec still trails TF-IDF by about 2–3 points (compare with Table 1). This suggests that linear SVMs align better with sparse, high-dimensional TF-IDF features, whereas document vectors from averaged Word2Vec models can lose frequency and contextual cues. To close the gap, consider TF-IDF-weighted averaging, tuning vector_size/window/min_count, or using non-linear classifiers.

3.3 Discussion

Overall, all configurations show good discriminative power (AUC 0.89–0.93) on data balanced with SMOTE. The two strongest determinants of performance are feature representation and hyperparameter tuning for the linear SVM. The best overall result is TF-IDF + linear SVM + Optuna.

1. TF-IDF + Optuna. As shown in Fig. 3, the model records 394 TN and 395 TP with 53 FP and 43 FN, indicating a slightly assertive positive prediction pattern. Fig. 4 confirms balanced per-class metrics (F1 ≈ 0.87; accuracy ≈ 86.7%). The ROC in Fig. 5 (AUC ≈ 0.93–0.94) shows strong separability and robustness to threshold changes.
2. Word2Vec + Random Search. As in Fig. 6, FN > FP, so the model is more conservative for positives; Fig. 7 reports accuracy ≈ 0.848 and F1 ≈ 0.84; Fig. 8 shows AUC ≈ 0.91—good, but below TF-IDF. Performance can be improved by adjusting thresholds or further tuning vector_size, window, and weighting.
3. Across features. Comparing Table 1 and Table 2, TF-IDF leads by roughly 2–3 points in accuracy/F1 with linear SVM (86.68%, F1 0.87 vs. 84.38%, F1 0.84). Linear SVM is naturally well-suited to sparse TF-IDF vectors, whereas averaged Word2Vec compresses frequency/context diversity, narrowing the margin.
4. Across tuning methods. With TF-IDF, Optuna is consistently best (86.68%, F1 0.87), followed by Random Search and Grid Search. With Word2Vec, Random Search slightly edges Optuna and Grid. This pattern is expected: Grid may miss good regions; Random explores large spaces efficiently; Optuna's TPE + pruning is most effective when the validation signal is clear (as with TF-IDF).

Because AUC is similar across top settings (about 0.92–0.93), gains primarily come from better C/loss/class_weight choices that balance precision and recall rather than from changes in global separability.

Practical implications. For deployment with a linear SVM, TF-IDF + Optuna is recommended for its highest and most stable performance. When computing is limited, Random Search offers a strong, lower-cost alternative. In scenarios requiring higher positive recall, adjust the threshold near the Youden's J maximum or calibrate scores to align with business objectives.

CONCLUSION

This study shows that the performance of linear SVM for sentiment classification is strongly affected by feature representation and the hyperparameter tuning strategy. TF IDF provides the best overall results, with TF IDF combined with linear SVM and Optuna achieving 86.68 percent accuracy, an F1 score of 0.87, and a ROC AUC of approximately 0.93-0.94, outperforming all other evaluated configurations. The best Word2Vec setting obtained via Random Search remains around 2 to 3 points lower, indicating limitations of simple averaged embedding representations when paired with a linear-margin classifier. Optuna yields the most consistent improvements for TF-IDF due to its adaptive TPE-based search and pruning, while Random Search offers a computationally efficient alternative with competitive performance. Applying SMOTE to the training set helps balance the class distribution and stabilizes macro-level metrics without a noticeable reduction in ROC AUC. These findings are limited to binary classification, a linear SVM classifier, and the dataset used in this study. Future work should examine multi-class and aspect-level settings, richer document representations, broader hyperparameter sensitivity analyses, and comparisons with nonlinear models and contextual embedding approaches to improve generalization and practical deployment.

Author Contributions

Fauziah Hanum: Conceptualization, Formal Analysis, Investigation, Writing-Original Draft. Richi Andrianto: Formal Analysis, Visualization, Writing-Review, and Editing. Anita Sri Rejeki Hutagaol: Funding Acquisition, Project Administration, Resources, Supervision, Validation, Writing- Review and Editing. Nurhanna Harahap: Data Curation, Investigation, Software, Writing- Review and Editing. Ibnu Rasyid Munthe: Methodology, Resources, Supervision, Writing- Review and Editing. All authors jointly discussed the research findings and contributed to the preparation of the final manuscript.

Funding Statement

This research was funded by the Directorate General of Higher Education, Research, and Technology (Kemdiktisaintek) under the Fundamental Research Scheme, Fiscal Year 2025.

Acknowledgment

The authors express their deepest gratitude to Universitas Labuhan Batu for the institutional support provided throughout the research process. Special thanks are also extended to the Directorate General of Higher Education, Research, and Technology (Kemdiktisaintek) for funding this research under the Fundamental Research Scheme, Fiscal Year 2025.

Declarations

The authors declare no competing interests.

Declaration of Generative AI and AI-assisted technologies

ChatGPT was used only as a language-editing aid. The authors confirm that no AI-generated content was incorporated without substantial author revision, and all statements were verified against the study's data and objectives.

REFERENCES

[1] S. Fernandes, R. Panda, V. G. Venkatesh, B. N. Swar, and Y. Shi, "MEASURING THE IMPACT OF ONLINE REVIEWS ON CONSUMER PURCHASE DECISIONS – A SCALE DEVELOPMENT STUDY," *Journal of Retailing and Consumer Services*, vol. 68, pp. 1–11, Sep. 2022, doi: https://doi.org/10.1016/j.jretconser.2022.103066

[2] D. M. Brown, S. Pattinson, C. Sutherland, and M. A. P. Davies, "INTERNAL MARKETING AND ORGANIZATIONAL PERFORMANCE: A SYSTEMATIC REVIEW AND FUTURE RESEARCH AGENDA," *J Bus Res*, vol. 194, pp. 1–14, May 2025, doi: https://doi.org/10.1016/j.jbusres.2025.115384

[3] M. K. Anam, T. P. Lestari, H. Yenni, T. Nasution, and M. B. Firdaus, "ENHANCEMENT OF MACHINE LEARNING ALGORITHM IN FINE-GRAINED SENTIMENT ANALYSIS USING THE ENSEMBLE," *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, vol. 19, no. 2, pp. 159–167, Mar. 2025, doi: https://doi.org/10.37936/ecti-cit.2025192.257815

[4] M. K. Anam *et al.*, "SARA DETECTION ON SOCIAL MEDIA USING DEEP LEARNING ALGORITHM DEVELOPMENT," *Journal of Applied Engineering and Technological Science*, vol. 6, no. 1, pp. 225–237, Dec. 2024, doi: https://doi.org/10.37385/jaets.v6i1.5390

[5] N. W. Susanto and H. Suparwito, "SVM-PSO ALGORITHM FOR TWEET SENTIMENT ANALYSIS #BESOKSENIN," *Indonesian Journal of Information Systems (IJIS)*, vol. 6, no. 1, pp. 36–47, 2023, doi: https://doi.org/10.24002/ijis.v6i1.7551

[6] M. A. K. Raiaan *et al.*, "A SYSTEMATIC REVIEW OF HYPERPARAMETER OPTIMIZATION TECHNIQUES IN CONVOLUTIONAL NEURAL NETWORKS," *Decision Analytics Journal*, vol. 11, pp. 1–32, Jun. 2024, doi: https://doi.org/10.1016/j.dajour.2024.100470

[7] M. Zainottah, R. Saputra, Y. Servanda, and I. Rosita, "CRITICAL SENTIMENT ANALYSIS OF TOKOPEDIA ELECTRONIC PRODUCTS USING SVM-LOGISTIC & TF-IDF ENSEMBLE METHODS," vol. 4, no. 3, pp. 2808–4519, 2025, doi: https://doi.org/10.59934/jaiea.v4i3.1194

[8] N. Z. B. Jannah and K. Kusnawi, "COMPARISON OF NAÏVE BAYES AND SVM IN SENTIMENT ANALYSIS OF PRODUCT REVIEWS ON MARKETPLACES," *Sinkron*, vol. 8, no. 2, pp. 727–733, Mar. 2024, doi: https://doi.org/10.33395/sinkron.v8i2.13559

[9] M. Hidayat and A. Wibowo, "SVM OPTIMIZATION WITH INFORMATION GAIN FEATURE SELECTION TO INCREASE THE ACCURACY OF SENTIMENT ANALYSIS OF INCREASING THE COST OF THE HAJJ," *Jurnal Teknik Informatika (Jutif)*, vol. 5, no. 4, pp. 579–591, Aug. 2024. , doi: https://doi.org/10.52436/1.jutif.2024.5.4.2217

[10] F. Farasalsabila *et al.*, "SENTIMENT ANALYSIS FOR IMDB MOVIE REVIEW USING SUPPORT VECTOR MACHINE (SVM) METHOD," *Inform: Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi*, vol. 8, no. 2, pp. 90–95, Mar. 2023, doi: https://doi.org/10.25139/inform.v8i2.5700

[11] F. Panjaitan, W. Ce, H. Oktafiandi, G. Kanugrahan, Y. Ramdhani, and V. H. C. Putra, "EVALUATION OF MACHINE LEARNING MODELS FOR SENTIMENT ANALYSIS IN THE SOUTH SUMATRA GOVERNOR ELECTION USING DATA BALANCING TECHNIQUES," *Journal of Information Systems and Informatics*, vol. 7, no. 1, pp. 461–478, Mar. 2025, doi: https://doi.org/10.51519/journalisi.v7i1.1019

[12] E. Elhosary and O. Moselhi, "EVALUATING NATURAL LANGUAGE PROCESSING ALGORITHMS FOR IMPROVED HAZARD AND OPERABILITY ANALYSIS," *Geodata and AI*, vol. 4, pp. 1–15, Sep. 2025, doi: https://doi.org/10.1016/j.geoai.2025.100026

[13] D. S. Asudani, N. K. Nagwani, and P. Singh, "IMPACT OF WORD EMBEDDING MODELS ON TEXT ANALYTICS IN DEEP LEARNING ENVIRONMENT: A REVIEW," *Artif Intell Rev*, vol. 56, no. 9, pp. 10345–10425, Sep. 2023, doi: https://doi.org/10.1007/s10462-023-10419-1

[14] A. S. Aribowo, N. H. Cahyana, and Y. Fauziah, "ENHANCING SEMI-SUPERVISED SENTIMENT ANALYSIS THROUGH HYPERPARAMETER TUNING WITHIN ITERATIONS: A COMPARATIVE STUDY USING GRID SEARCH AND RANDOM SEARCH," in *International Conference on Advanced Informatics and Intelligent Information Systems*, pp. 248–260, 2024, doi: https://doi.org/10.2991/978-94-6463-366-5_23

[15] A. A. R. Reza and M. S. Rohman, "PREDICTION STUNTING ANALYSIS USING RANDOM FOREST ALGORITHM AND RANDOM SEARCH OPTIMIZATION," *JOURNAL OF INFORMATICS AND TELECOMMUNICATION ENGINEERING*, vol. 7, no. 2, pp. 534–544, Jan. 2024, doi: https://doi.org/10.31289/jite.v7i2.10628

[16] Herianto, B. Kurniawan, Z. H. Hartomi, Y. Irawan, and M. K. Anam, "MACHINE LEARNING ALGORITHM OPTIMIZATION USING STACKING TECHNIQUE FOR GRADUATION PREDICTION," *Journal of Applied Data Sciences*, vol. 5, no. 3, pp. 1272–1285, Sep. 2024, doi: https://doi.org/10.47738/jads.v5i3.316

[17] M. K. Anam *et al.*, "IMPROVED PERFORMANCE OF HYBRID GRU-BILSTM FOR DETECTION EMOTION ON TWITTER DATASET," *Journal of Applied Data Sciences*, vol. 6, no. 1, pp. 354–365, Jan. 2025, doi: https://doi.org/10.47738/jads.v6i1.459

[18] Y. D. Nugroho H, F. Zakiyabarsi, and A. J. Paramita, "IMPLEMENTASI SMOTE-ENN DAN BORDERLINE SMOTE TERHADAP PERFORMA LIGHTGBM PADA IMBALANCED CLASS," *Rabit: Jurnal Teknologi dan Sistem Informasi Univrab*, vol. 10, no. 1, pp. 51–59, Jan. 2025, doi: https://doi.org/10.36341/rabit.v10i1.5436

[19] M. A. Latief, L. R. Nabila, W. Miftakhurrahman, S. Ma'rufatullah, and H. Tantyoko, "HANDLING IMBALANCE DATA USING HYBRID SAMPLING SMOTE-ENN IN LUNG CANCER CLASSIFICATION," *International Journal of Engineering and Computer Science Applications (IJECSA)*, vol. 3, no. 1, pp. 11–18, Feb. 2024, doi: https://doi.org/10.30812/ijecsa.v3i1.3758

[20] P. P. Putra, M. K. Anam, A. S. Chan, A. Hadi, N. Hendri, and A. Masnur, "OPTIMIZING SENTIMENT ANALYSIS ON IMBALANCED HOTEL REVIEW DATA USING SMOTE AND ENSEMBLE MACHINE LEARNING TECHNIQUES," *Journal of Applied Data Sciences*, vol. 6, no. 2, pp. 936–951, May 2025, doi: https://doi.org/10.47738/jads.v6i2.618

[21] F. Suandi *et al.*, "ENHANCING SENTIMENT ANALYSIS PERFORMANCE USING SMOTE AND MAJORITY VOTING IN MACHINE LEARNING ALGORITHMS," in *International Conference on Applied Engineering*, Atlantis Press, 2024, pp. 126–138, doi: https://doi.org/10.2991/978-94-6463-620-8_10

[22] M. K. Anam, T. P. Lestari, L. Efrizoni, N. S. Handayani, and I. Andhika, "SENTIMENT ANALYSIS OPTIMIZATION USING ENSEMBLE OF MULTIPLE SVM KERNEL FUNCTIONS," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 9, no. 4, pp. 905–914, Aug. 2025, doi: https://doi.org/10.29207/resti.v9i4.6708

[23] V. Talasila, M. V Mohan, and N. M. R, "ENHANCING TEXT-TO-IMAGE SYNTHESIS WITH AN IMPROVED SEMI-SUPERVISED IMAGE GENERATION MODEL INCORPORATING N-GRAM, ENHANCED TF-IDF, AND BOW TECHNIQUES," *International Journal of Intelligent Systems and Applications in Engineering* , vol. 11, no. 7s, pp. 381–397, 2023, [Online]. Available: www.ijisae.org

[24] B. Kabra and C. Nagar, "CONVOLUTIONAL NEURAL NETWORK BASED SENTIMENT ANALYSIS WITH TF-IDF BASED VECTORIZATION," *Pg 1 J. Integr. Sci. Technol*, vol. 11, no. 3, p. 503, 2023, [Online]. Available: http://pubs.thesciencein.org/jist

[25] H. Zhou, "RESEARCH OF TEXT CLASSIFICATION BASED ON TF-IDF AND CNN-LSTM," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, pp. 1–8, Jan. 2022, doi: https://doi.org/10.1088/1742-6596/2171/1/012021

[26] S. A. Savittri, A. Amalia, and M. A. Budiman, "A RELEVANT DOCUMENT SEARCH SYSTEM MODEL USING WORD2VEC APPROACHES," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, pp. 1–9, Jun. 2021, , doi: https://doi.org/10.1088/1742-6596/1898/1/012008

[27] Sugiyanti and A. Fricco, "SHOE REVIEW SENTIMENT ANALYSIS USING MACHINE LEARNING AND DEEP LEARNING WITH WORD2VEC," *Jurnal Elektronik Ilmu Komputer Udayana*, vol. 12, no. 1, pp. 215–222, 2023, doi: https://doi.org/10.24843/JLK.2023.v12.i01.p25

[28] M. Lestandy and Abdurrahim, "EFFECT OF WORD2VEC WEIGHTING WITH CNN-BILSTM MODEL ON EMOTION CLASSIFICATION," *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, vol. 12, no. 1, pp. 99–107, Mar. 2023, doi: https://doi.org/10.23887/janapati.v12i1.58571

[29] E. M. S. Rochman *et al.*, "CLASSIFICATION OF THESIS TOPICS BASED ON INFORMATICS SCIENCE USING SVM," in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, pp. 1–7, May 2021, doi: https://doi.org/10.1088/1757-899X/1125/1/012033

[30] Sugiyanti and A. Fricco, "SHOE REVIEW SENTIMENT ANALYSIS USING MACHINE LEARNING AND DEEP LEARNING WITH WORD2VEC," *Jurnal Elektronik Ilmu Komputer Udayana*, vol. 12, no. 1, pp. 215–222, 2023, doi: https://doi.org/10.24843/JLK.2023.v12.i01.p25

[31] M. Lestandy and Abdurrahim, "EFFECT OF WORD2VEC WEIGHTING WITH CNN-BILSTM MODEL ON EMOTION CLASSIFICATION," *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, vol. 12, no. 1, pp. 99–107, Mar. 2023, doi: https://doi.org/10.23887/janapati.v12i1.58571