Journal of Applied Informatics and Computing (JAIC)
Vol.10, No.3, June 2026, pp. 3071~3083
e-ISSN: 2548-6861

From Sparse Features to Transformers: A Statistical Evaluation of TF-IDF, FastText, and IndoBERT for Sentiment Classification of Indonesian Travel App Reviews

Claudian Tikulimbong Tangdilomban [1]*, Syaifullah Yusuf Ramdhan [2]*, Muhammad Rizal [3]*, Cici Suhaeni [4]*, Bagus Sartono [5]*
*School of Data Science, Mathematics and Informatics, IPB University, Indonesia
claudian_tangdilomban@apps.ipb.ac.id [1], syaifullahramadhan@apps.ipb.ac.id [2], mmmdrizal@apps.ipb.ac.id [3], cici_suhaeni@apps.ipb.ac.id[4], bagusco@apps.ipb.ac.id [5]

Article Info

Article history:

Received 2026-03-02
Revised 2026-06-16
Accepted 2026-06-18

Keyword:

Sentiment Analysis,
TF-IDF,
FastText,
IndoBERT,
Travel Application Reviews.

ABSTRACT

This study compares three text representation techniques, namely TF-IDF, FastText, and IndoBERT, in the sentiment classification task of Indonesian-language user reviews of travel applications. The dataset consists of 4.000 reviews from Traveloka and Tiket.com, collected through Google Play Store scraping and manually annotated with sentiment labels. Each representation technique was combined with three classification algorithms, namely Support Vector Machine, Logistic Regression, and Random Forest, resulting in nine experimental configurations. The evaluation was conducted using stratified 5-fold cross-validation with macro F1-score as the primary metric, supported by hyperparameter tuning using GridSearchCV, paired t-test statistical analysis, and Cohen's d effect size measurement. The evaluation results indicate that IndoBERT generally achieved the best performance compared to TF-IDF and FastText. The best configuration was obtained by IndoBERT with Logistic Regression, achieving an F1-score of 0.9261 after tuning. The statistical test showed that the performance differences among text representations were statistically significant, with large effect sizes in the comparison between IndoBERT and TF-IDF ($d = -1.36$) and between IndoBERT and FastText ($d = -1.10$). Nevertheless, TF-IDF combined with Logistic Regression and SVM remained competitive, achieving an F1-score of approximately 0.892 after tuning, making it a lightweight and interpretable alternative. This study concludes that the quality of text representation has a more dominant influence on sentiment classification performance than the complexity of the classification algorithm.

I. INTRODUCTION

The tourism and hospitality sector is one of the key components driving the growth of the digital economy in Indonesia. Digital transformation has changed consumer behavior, particularly in the way people plan trips, select services, and book transportation and accommodation online. Platforms such as Traveloka and Tiket.com have become widely used services because they offer ease of access and integrated features. As the number of users continues to increase, the volume of generated reviews also grows, making them a rich yet complex source of data to analyze [1].

User reviews contain valuable information that reflects users' experiences, satisfaction, and complaints regarding the services they use [2]. In the context of travel applications, user reviews and ratings can influence the trust of prospective customers while also serving as a basis for data-driven service improvement [3]. However, review data are unstructured and are generally presented in short texts, often using informal language, non-standard words, regional dialects, and colloquial expressions, making them difficult to analyze manually or using conventional methods [2]. Manually processing large volumes of data is also inefficient and time-

consuming, making automated approaches increasingly necessary [4].

Sentiment analysis offers a solution by extracting user opinions through polarity classification, such as positive and negative sentiment. This approach has been widely applied in e-commerce and public service domains to evaluate consumer perceptions [5], including reviews of digital applications such as Trivago, Tiket, Booking, Traveloka, and Agoda using algorithms such as SVM and Random Forest [6].

A crucial stage in sentiment analysis is transforming text into numerical representations. TF-IDF (Term Frequency–Inverse Document Frequency) is one of the most commonly used techniques, designed to measure the importance of a word in a document relative to the entire corpus (Ramos, 2003). This approach is effective for high-dimensional data and remains stable across various text classification scenarios, particularly when combined with SVM [7]. However, bag-of-ngrams-based TF-IDF has limitations in understanding contrastive sentences, informal language, and contextual nuances [8].

As an alternative, FastText offers a neural network-based embedding approach that utilizes subword information through character n-grams, enabling it to generate representations for rare words and word-form variations [9]. FastText is widely used because it has the potential to handle non-standard words and informal language using subword information [10], [11]. Nevertheless, its effectiveness is not always superior to that of TF-IDF. [12] reported that TF-IDF achieved an accuracy of 71%, compared with FastText, which achieved only 50% in sentiment classification using SVM. This finding indicates that embedding-based representations are not automatically better than frequency-based representations.

These varied findings indicate that the performance of each representation method can strongly depend on the characteristics of the data domain [13], [14]. Therefore, comparative evaluation in a specific domain remains necessary. In addition to these two approaches, the development of transformer-based models has introduced a new dimension in contextual text representation. BERT (Bidirectional Encoder Representations from Transformers) produces bidirectional representations by considering both left and right contexts simultaneously [15] and serves as the foundation for IndoBERT, a monolingual BERT model for the Indonesian language trained on Indonesian corpora, which has demonstrated strong performance across various NLP tasks in the IndoLEM benchmark [16]. IndoBERT is relevant for Indonesian sentiment classification because it can capture language complexity, including informal structures and linguistic [2], [17], and has been applied in aspect-based sentiment analysis of tourism destination reviews [18].

Although these three text representation approaches have been widely studied, existing comparative studies generally remain focused on descriptive metrics without incorporating formal statistical inference. Aspects such as variability across folds, statistical significance testing, and effect size measurement of performance differences are still rarely discussed in depth. As a result, it remains difficult to determine whether the observed performance differences are truly statistically meaningful, particularly in the context of short Indonesian-language review texts that are specific to a particular domain.

Based on this gap, this study develops and evaluates a sentiment classification system for user reviews on the Traveloka and Tiket.com platforms by comparing three text representation techniques, namely TF-IDF, FastText, and IndoBERT. Three supervised learning algorithms, namely Support Vector Machine (SVM), Logistic Regression (LR), and Random Forest (RF), are used as classification methods. The evaluation is conducted using accuracy, precision, recall, and F1-score metrics, accompanied by cross-validation and statistical testing to assess the stability and significance of performance differences among text representations.

Empirically, this study contributes by comparing frequency-based, static embedding-based, and contextual transformer-based representations in the domain of Indonesian travel application reviews. Unlike previous studies that primarily compare TF-IDF and FastText using descriptive metrics, this study contributes by incorporating IndoBERT as a contextual transformer-based baseline, employing paired t-test with Cohen's d effect size to assess statistical significance of performance differences, and providing error analysis specific to Indonesian-language travel application reviews.Practically, the findings of this study are expected to provide insights into the most suitable text representations and classifiers for analyzing user reviews on Traveloka and Tiket.com, thereby supporting complaint monitoring, service feature evaluation, and data-driven decision-making in digital travel platforms.

II. METHODOLOGY

A. Data and Labeling

The research data were obtained through the process of scraping user reviews from the Google Play Store using the Python library google-play-scraper. This study focuses on two popular travel applications in Indonesia, namely Traveloka and Tiket.com. The final dataset used in this study consists of 4.000 reviews, with each application contributing 2.000 reviews. The collected data include review text, application name, review date, star rating, and sentiment label. Sentiment labeling was conducted semi-automatically through a combination of star ratings and manual validation of the review content. In the initial stage, reviews with ratings of 4 and 5 were categorized as positive sentiment, while reviews with ratings of 1 and 2 were categorized as negative sentiment. Reviews with a rating of 3 were excluded from the dataset because they tend to be neutral or ambiguous, which may obscure the boundary between positive and negative classes. To improve label reliability, the initial rating-based labels were then manually re-examined by two researchers. Both researchers read the review content and evaluated whether the rating-based labels were consistent with the sentiment polarity expressed in the text. If inconsistencies

were found between the rating and the review content, the label was adjusted based on the sentiment meaning conveyed in the text. Differences in judgment between the two researchers were resolved through discussion until a final agreement was reached. Thus, the labels used in the experiments are manually validated labels, rather than labels derived solely from star ratings.

In addition to label validation, this study also examined the potential presence of duplicate reviews and class distribution. Duplicate checking was performed based on the combination of application name, review text, and sentiment label. Class distribution was analyzed to determine the proportion of positive and negative reviews and to calculate the imbalance ratio. This analysis is important because class imbalance may affect the interpretation of model performance, particularly when using accuracy as a metric. Therefore, F1-score was used as one of the main evaluation metrics because it represents the balance between precision and recall.

B. Text Preprocessing

Text preprocessing is an important stage in transforming raw text data into a more structured format that is ready to be represented numerically. However, the preprocessing steps in this study were not applied identically to all text representations because TF-IDF, FastText, and IndoBERT have different characteristics. The first stage was emoticon normalization. Emoticons in the text, such as 😊 or 😔, were normalized into Indonesian words with equivalent semantic meanings, such as "senang" and "sedih", using a Unicode-emoticon mapping dictionary. This step was performed before character cleaning so that emoticon symbols were not removed before being converted into textual information [19]. The second stage was text cleaning. All text was converted to lowercase to avoid feature duplication caused by differences in capitalization. Furthermore, non-alphabetic characters, symbols, numbers, punctuation marks, and excessive spaces were removed using regular expressions to produce more standardized and noise-free text [20]. The third stage was tokenization. Tokenization was performed simply by splitting the text based on spaces after the text-cleaning process. This tokenization was used to support the normalization of non-standard words, stopword removal, and the construction of word-based representations. Meanwhile, the Sastrawi library was used specifically in the stemming stage, not as the main tokenizer. The fourth stage was the normalization of non-standard words. Non-standard words commonly found in user reviews were normalized using an Indonesian slang dictionary. For example, informal spelling variations such as "gk", "ga", or "nggak" were normalized into "tidak", while words such as "mantul" were normalized into "bagus". This stage aimed to reduce orthographic variations that do not add new semantic information, thereby making the text representation more consistent. The fifth stage was stopword removal. Common words with little or no sentiment-bearing meaning, such as "yang", "dengan", and "di", were removed using an Indonesian stopword list that had been adjusted to the context of application reviews. This stage was carried out to reduce less informative features and help the model focus more on words that contribute to sentiment [21]. The sixth stage was stemming. Stemming was performed using the Sastrawi algorithm to transform affixed words into their root forms. For example, the words "menyukai", "disukai", and "kesukaan" were reduced to the root word "suka". This stage was applied to the TF-IDF representation because TF-IDF operates based on token frequency, so unifying word-form variations can help reduce feature sparsity.

In this study, the preprocessing pipeline was differentiated according to the type of text representation. For TF-IDF, the preprocessing stages included emoticon normalization, text cleaning, normalization of non-standard words, stopword removal, and stemming. For FastText, the preprocessing stages were applied up to stopword removal without stemming. This was done because FastText utilizes subword information through character n-grams, so preserving the original word forms can retain useful morphological information. Meanwhile, for IndoBERT, preprocessing was kept more minimal, consisting of emoticon normalization and light cleaning without stemming, stopword removal, or slang normalization. This decision was made because transformer-based models require relatively intact sentence context so that the resulting contextual representations remain optimal.

C. Text Representation

After preprocessing, the review texts were transformed into numerical form using three representation approaches, namely TF-IDF, FastText, and IndoBERT.

1. TF-IDF

TF-IDF (Term Frequency–Inverse Document Frequency) calculates the importance weight of a word in a document relative to the entire corpus [22]. Words that appear frequently in a particular document but rarely appear in other documents receive higher weights. This approach produces a sparse feature matrix and is widely used in text classification because it is simple, efficient, and easy to interpret.

In this study, TF-IDF was constructed using unigrams and bigrams with a maximum of 5.000 features. The use of bigrams aimed to capture word pairs that carry specific sentiment meanings, such as "tidak bisa", "sangat buruk", or "mudah digunakan". To prevent data leakage, the TF-IDF feature construction process was performed within the cross-validation pipeline. Thus, the vectorizer was fitted only on the training data in each fold and did not use information from the validation data.

2. FastText

FastText produces dense vectors by considering subword information through character n-grams, enabling it to generate representations for rare words and word-form variations that may not always appear in the training data [9]. This advantage makes FastText relevant for Indonesian review texts, which often contain non-standard words, spelling variations, and informal forms. The FastText embeddings in this study were obtained using the pre-trained Indonesian model cc.id.300, which produces 300-

dimensional vectors. The pre-trained model cc.id.300 was trained with the following default configuration: vector dimension of 300, window size of 5, minimum word count of 5, and character n-gram range of 3 to 6. No additional training was performed on the research dataset. Document-level representations were obtained by calculating the average word vectors of all tokens in each review. If a review consisted of several words, each word was converted into a FastText vector, and all word vectors were then averaged to produce a single document vector. Because the FastText model used in this study was pre-trained and was not retrained on the research dataset, embedding extraction was performed before the cross-validation process without causing data leakage.

3. IndoBERT

IndoBERT is a monolingual BERT model for the Indonesian language trained on large-scale corpora and has demonstrated superior performance across various NLP tasks in the IndoLEM benchmark [16]. Unlike TF-IDF and FastText, IndoBERT produces contextual representations that consider word meaning based on the overall sentence context [15]. This representation is relevant for review texts because the same word may have different meanings depending on its usage context.

In this study, IndoBERT was used as a frozen feature extractor, not through a fine-tuning process. The model was loaded in inference mode or eval mode without weight updates, so the IndoBERT parameters remained fixed during feature extraction. Each review was processed using the indobenchmark/indobert-base-p1 model with a maximum token length of 128 and a batch size of 32. Document representations were obtained from the [CLS] token vector in the last hidden state, producing a 768-dimensional vector for each review. This vector was then used as input features for the SVM, Logistic Regression, and Random Forest classification models, in the same way that TF-IDF and FastText features were used.

D. Classification Models

The three text representations, namely TF-IDF, FastText, and IndoBERT, produce numerical vectors that are subsequently used as input features for three supervised learning algorithms, namely Support Vector Machine, Logistic Regression, and Random Forest. The selection of these three models was based on their widespread use in text classification tasks and their ability to handle high-dimensional features.

1. Support Vector Machine (SVM)

Support Vector Machine (SVM) constructs a separating hyperplane in a high-dimensional feature space with the objective of maximizing the margin between two classes [23]. The SVM classification function can be expressed as:

$$f(x) = sign(w^T x + b)$$

where w denotes the weight vector, x denotes the feature vector, and b denotes the bias. The model is optimized to obtain a hyperplane that can separate positive and negative classes with the maximum margin. SVM is known to perform strongly on text data because of its ability to handle sparse and high-dimensional features.

2. Logistic Regression (LR)

Logistic Regression (LR) is a linear classification method that estimates the probability of class membership using the sigmoid function [24]:

$$P(y = 1|x) = \frac{1}{1 + e^{-(w^T x + b)}}$$

Although it is considered a simple model, LR demonstrates stable performance, computational efficiency, and ease of interpretation. Therefore, LR is often used as a strong comparative model in text classification experiments [25], [26].

3. Random Forest (RF)

Random Forest (RF) is an ensemble learning technique that combines the prediction results of multiple decision trees through a majority voting mechanism [27]:

$$\hat{y} = majorityvote(h_1(x), h_2(x), \ldots, h_n(x))$$

where $h_i(x)$ represents the prediction of the i-th decision tree for data x. This approach can reduce the risk of overfitting and improve model robustness against noise in the data. In the context of text classification, RF was used to examine whether a non-linear approach could provide competitive performance compared to linear models [28].

In addition to the three main classifiers, this study also used DummyClassifier as a baseline. DummyClassifier applies a majority-class prediction strategy and functions as a minimum performance benchmark. With this baseline, the performance of the main models can be evaluated to determine whether they truly outperform simple predictions based on class distribution.

E. Experimental Procedure

The experimental evaluation was conducted using stratified 5-fold cross-validation to ensure that class distribution remained proportional in each fold and to reduce the dependence of the results on a single data split. In this procedure, the dataset was divided into five subsets. In each iteration, one subset was used as validation data and the remaining four subsets were used as training data. This process was repeated five times, so that each observation was used exactly once as validation data [29].

The combination of three text representations, namely TF-IDF, FastText, and IndoBERT, with three main classification models, namely SVM, LR, and RF, resulted in a total of nine experimental configurations. In addition, DummyClassifier was used as a baseline to provide a minimum performance reference. For the TF-IDF representation, feature construction was performed using a pipeline so that the vectorizer fitting process was carried out only on the training data in each fold. This is important to avoid data leakage from validation data into the feature construction process. For FastText and IndoBERT, embedding extraction was performed before cross-validation because both used pre-trained models and did not learn new parameters from the research dataset. After the

embeddings were generated, the vectors were partitioned according to the fold split in each cross-validation iteration.

In addition to evaluation using default parameters, this study also performed hyperparameter tuning using GridSearchCV. For TF-IDF, the tested parameters included combinations of n-grams, maximum number of features, regularization values, class weight, and Random Forest parameters. For FastText and IndoBERT, tuning was performed on the classifier parameters because the embeddings had already been extracted using pre-trained models.

F. Evaluation and Statistical Testing

The performance of each experimental configuration was evaluated using four standard classification metrics, namely accuracy, precision, recall, and F1-score. These metrics were calculated based on true positive (TP), true negative (TN), false positive (FP), and false negative (FN) values [30]. Accuracy was used to measure the proportion of correct predictions among all data. Precision was used to measure the correctness of positive predictions, while recall was used to measure the model's ability to identify positive data. F1-score was used as the main metric because it combines precision and recall, making it more informative for data with potential class imbalance.

The value of each metric was calculated for each fold and then averaged to obtain the final score. In addition to the mean value, the standard deviation across folds was also reported to indicate model performance stability. A model with a high F1-score and a low standard deviation was considered to have better and more stable performance.

To assess the statistical significance of performance differences among configurations, a paired t-test was conducted on the F1-score values from each fold. This test was used because the performance of two models was compared on the same folds, making the observations paired. In addition to the p-value, this study also calculated Cohen's d to measure the effect size of performance differences. Since the number of folds in cross-validation is limited, the results of the statistical tests were interpreted carefully as an indication of performance differences, rather than as the sole basis for drawing conclusions.

III. Result and Discussion

This study was conducted to evaluate and compare the effectiveness of three text representation methods TF-IDF, FastText and IndoBERT for sentiment classification of Indonesian-language user reviews on the Tiket.com and Traveloka applications. The supervised learning models employed include Support Vector Machine (SVM), Logistic Regression (LR), and Random Forest (RF). Evaluation was carried out using accuracy, precision, recall, and F1-score as performance metrics.

A. Dataset Characteristics and Sentiment Distribution

The dataset used in this study consists of 4,000 user reviews of the Traveloka and Tiket.com applications obtained from the Google Play Store. The data had undergone a manual labeling process and were classified into two sentiment classes, namely positive and negative. Based on the label distribution, there were 2.120 negative reviews, accounting for 53,0%, and 1.880 positive reviews, accounting for 47,0%. Thus, the number of negative reviews was slightly higher than the number of positive reviews, although the difference was not substantial. The distribution of sentiment labels is presented in Table I.

TABLE I
DISTRIBUTION OF SENTIMENT LABELS

Sentiment Label	Number of Reviews
Negative	2.120
Positive	1.880
Total	4.000

Based on Table I, an imbalance ratio of 1.13:1 was obtained. This value indicates that the dataset falls into the balanced category. Under this condition, model performance is not overly influenced by the dominance of one class. Nevertheless, this study still uses F1-score as the main evaluation metric in addition to accuracy, precision, and recall. F1-score was used because it provides a more proportional representation of the model's ability to identify both sentiment classes. The distribution of positive and negative reviews is visualized in Figure 1.

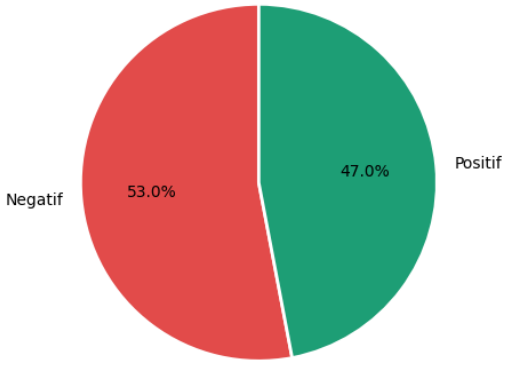

Figure 1. Distribution and proportion of manual sentiment labels

To evaluate the reliability of the sentiment labels, inter-annotator agreement was measured using Cohen's Kappa between the manually validated labels and the second annotator labels. The resulting Cohen's Kappa value was 0.9736, indicating almost perfect agreement. This result confirms that the ground-truth labels used for model training and evaluation were highly reliable.

In addition to label distribution, word characteristics in each class were analyzed using word clouds. The words cloud visualization for positive and negative reviews is presented in Figure 2. In positive reviews, the dominant words included "mantap", "bagus", "aplikasi", "mudah", "oke", "cepat", "murah", "promo", and "terima kasih". These words indicate that positive reviews are generally related to ease of application use, satisfaction with services, prices or promotions, and transaction experiences that users perceived as helpful.

In contrast, the dominant words in negative reviews included “refund”, “bayar”, “hotel”, “kecewa”, “harga”, “tiket”, “pesawat”, “susah”, “gagal”, and “error”. These words show that user complaints were mostly related to payment processes, refunds, ticket or hotel bookings, technical problems in the application, and service experiences that did not meet expectations. The dominance of the word “refund” in negative reviews indicates that refund issues are one of the important aspects of negative user experiences in travel applications.

Figure 2. Word cloud of positive and negative reviews based on manual labels

Overall, the exploratory analysis results show that the two sentiment classes have fairly distinct word characteristics. Positive reviews tend to contain words indicating satisfaction, ease of use, and the benefits of using the application, whereas negative reviews more frequently contain words related to service complaints, transaction problems, and technical issues. These differences in word patterns indicate that text representations such as TF-IDF, FastText, and IndoBERT have sufficient linguistic information to distinguish sentiment polarity in the travel application review dataset.

B. Performance of Default Models

The initial evaluation was conducted using default parameters for all combinations of text representations and classification models. The three text representations compared were TF-IDF, FastText, and IndoBERT, while the classification models used included Support Vector Machine (SVM), Logistic Regression (LR), and Random Forest (RF). In addition, DummyClassifier was used as a baseline to indicate the minimum performance when the model only follows the majority class.

The default evaluation results show that the IndoBERT representation achieved the highest performance compared to TF-IDF and FastText. The best combination under the default setting was obtained by IndoBERT with Random Forest, with an F1-score of 0,9152 and an accuracy of 0,9160. IndoBERT with Logistic Regression ranked second with an F1-score of 0,9086, followed by FastText with SVM with an F1-score of 0,8928. Meanwhile, DummyClassifier achieved only an F1-score of 0,3464, indicating that all main models performed far better than the simple baseline. The detailed performance of each representation and classifier combination is presented in Table II.

TABLE II
DEFAULT MODEL PERFORMANCE ACROSS ALL TEXT REPRESENTATIONS

Representation	Model	Mean F1	Std F1	Accuracy	Precision	Recall
IndoBERT	Random Forest	0,9152	0,0115	0,9160	0,9195	0,9135
	Logistic Regression	0,9086	0,0101	0,9090	0,9092	0,9086
	SVM	0,8894	0,0111	0,8898	0,8895	0,8896
FastText	Random Forest	0,8745	0,0136	0,8765	0,8847	0,8722
	Logistic Regression	0,8792	0,0143	0,8810	0,8883	0,8770
	SVM	0,8928	0,0118	0,8938	0,8968	0,8912
TF-IDF	Random Forest	0,8703	0,0100	0,8707	0,8703	0,8704
	Logistic Regression	0,8687	0,0300	0,8710	0,8798	0,8670
	SVM	0,8830	0,0101	0,8835	0,8833	0,8830

Based on Table II, IndoBERT outperformed the other representations in two out of three classifiers, namely Logistic Regression and Random Forest. This advantage indicates that the contextual representation generated by IndoBERT is able to capture sentiment information more effectively in Indonesian-language travel application reviews. This is particularly important because user reviews do not always contain explicit sentiment words, but may also include context, negation, or implicit complaints. This finding is consistent with the characteristics of BERT as a bidirectional model that considers both left and right contexts in representation learning, as well as IndoBERT as a model specifically developed for the Indonesian language [15], [16].

FastText showed competitive performance, particularly when combined with SVM. Its F1-score of 0,8928 indicates that subword-based representation was still able to capture variations in word forms within user reviews. Conceptually, FastText has an advantage through its use of character n-grams, which enables it to represent rare words or word variations that may not always appear in the training data [9]. However, in this study, its performance did not surpass IndoBERT.

Meanwhile, TF-IDF still demonstrated relatively good performance despite being a frequency-based representation. The combination of TF-IDF with SVM achieved an F1-score of 0,8830. This result indicates that in relatively short review texts, explicit words such as “bagus”, “mudah”, “refund”, “gagal”, and “error” remain highly informative for distinguishing positive and negative sentiments. Therefore, although IndoBERT achieved the best performance, TF-IDF can still be considered a lightweight and interpretable alternative.

C. Performance After Hyperparameter Tuning

After the evaluation using default parameters, hyperparameter tuning was conducted to obtain the best parameter configuration for each combination of text representation and classifier. The tuning process was carried out using GridSearchCV with macro F1-score as the optimization metric. This tuning aimed to examine whether parameter adjustment could improve model performance compared to the default setting.

The tuning results show that the combination of IndoBERT with Logistic Regression achieved the best overall performance, with an F1-score of 0,9261. The best parameters for this configuration were a regularization value of C = 0,1 and the use of class_weight = balanced. This result indicates that Logistic Regression was able to effectively utilize IndoBERT’s contextual representation, particularly in a high-dimensional feature space. The complete hyperparameter tuning results are presented in Table III.

TABLE III
HYPERPARAMETER TUNING RESULTS

Representation	Model	Best F1-score
IndoBERT	Logistic Regression	0,9261
IndoBERT	Random Forest	0,9160
IndoBERT	SVM	0,9041
FastText	Logistic Regression	0,8943
FastText	SVM	0,8928
TF-IDF	Logistic Regression	0,8927
TF-IDF	SVM	0,8926
FastText	Random Forest	0,8760
TF-IDF	Random Forest	0,8733

Based on Table III, tuning changed the best configuration from IndoBERT with Random Forest under the default setting to IndoBERT with Logistic Regression after parameter optimization. Under the default setting, IndoBERT with Random Forest achieved an F1-score of 0,9152, while after tuning, IndoBERT with Logistic Regression increased to 0,9261. This indicates that a linear model such as Logistic Regression can perform very well when the feature representation used is already informative and contextual.

Performance improvements were also observed in several other configurations. TF-IDF with Logistic Regression increased from 0,8687 to 0,8927 after tuning, while TF-IDF with SVM increased from 0,8830 to 0,8926. For FastText, Logistic Regression increased from 0,8792 to 0,8943. However, the improvement in FastText with SVM was relatively unchanged because its default configuration was already close to the best parameter setting. This finding highlights the importance of hyperparameter tuning in sentiment classification. A previous study on Traveloka reviews also showed that optimizing TF-IDF and SVM using Grid Search with stratified 5-fold cross-validation can produce a strong baseline for sentiment classification of travel application reviews [8]. In this study, tuning not only improved model performance but also clarified that parameter selection plays an important role in determining the final performance.

Interestingly, after tuning, the performance of TF-IDF with Logistic Regression and TF-IDF with SVM became almost equivalent to FastText. This indicates that in a relatively short travel application review dataset, frequency-based representation remains highly competitive when model parameters are properly optimized. The advantage of FastText as a subword-based embedding does not always result in a large performance improvement because the document representation was formed by averaging word vectors, meaning that context and word order were not fully preserved.

Overall, the tuning results reinforce the finding that IndoBERT is the best representation in this study, particularly when combined with Logistic Regression. However, for systems that require a lighter and more interpretable approach, TF-IDF with SVM or Logistic Regression can still be considered because it provides competitive performance with lower computational requirements.

D. Statistical Testing and Model Stability

Stability analysis was conducted to examine the consistency of model performance across each cross-validation fold. Based on the mean F1-score and standard deviation, IndoBERT-based models showed relatively higher and more stable performance compared to TF-IDF and FastText. In the Logistic Regression model, IndoBERT achieved a mean F1-score of 0,9086, with a standard deviation of 0,0101 and a coefficient of variation of 1,11%. This value indicates that the performance of IndoBERT with Logistic Regression was not only high but also relatively consistent across folds. The model stability results based on F1-score are presented in Table IV.

TABLE IV
MODEL PERFORMANCE STABILITY BASED ON F1-SCORE

Model	TF-IDF Mean	TF-IDF Std	TF-IDF CV (%)	FastText Mean	FastText Std	FastText CV (%)	IndoBERT Mean	IndoBERT Std	IndoBERT CV (%)
SVM	0,8830	0,0101	1,15	0,8928	0,0118	1,32	0,8894	0,0111	1,25
Logistic Regression	0,8687	0,0300	3,46	0,8792	0,0143	1,63	0,9086	0,0101	1,11
Random Forest	0,8703	0,0100	1,15	0,8745	0,0136	1,55	0,9152	0,0115	1,25

In the SVM model, the performance of the three representations was within a relatively close range. TF-IDF achieved a mean F1-score of 0,8830, FastText achieved 0,8928, and IndoBERT achieved 0,8894. This indicates that, in the SVM model, FastText slightly outperformed IndoBERT and TF-IDF under the default setting. However, in Logistic Regression and Random Forest, IndoBERT showed a clearer advantage. IndoBERT with Logistic Regression achieved an F1-score of 0,9086, while IndoBERT with Random Forest achieved an F1-score of 0,9152.

The learning curve analysis was also conducted to observe the learning patterns of each representation and classifier combination. The learning curve results are presented in Figure 3. In several configurations, the training score was higher than the validation score, particularly in Random Forest, indicating a tendency toward overfitting. In contrast, Logistic Regression with IndoBERT showed a more moderate gap between the training score and validation score, indicating better generalization. In general, increasing the amount of training data improved the validation score, especially in the FastText and TF-IDF combinations, although the improvement began to slow down at larger training sizes.

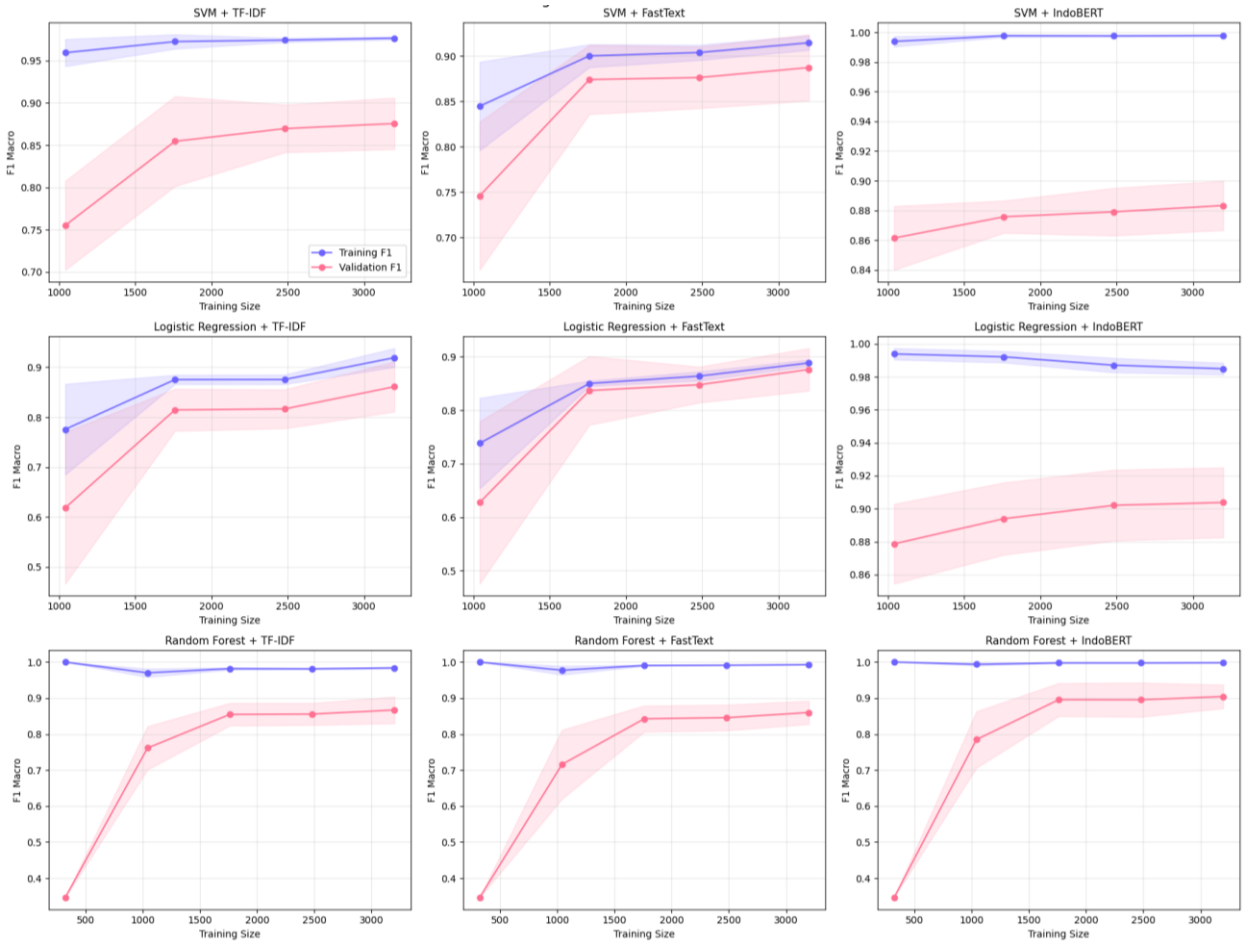

Figure 3. Learning curve of all combinations of representations and classifiers

In addition to descriptive stability analysis, a paired statistical test was conducted to compare the performance of text representations based on F1-score values obtained from the same cross-validation folds. The paired t-test was used because the comparisons were made on paired observations across identical folds. To support the interpretation, Cohen's d was also calculated to measure the magnitude of the performance differences. The statistical comparison results are presented in Table V.

TABLE V
STATISTICAL TEST FOR COMPARISON AMONG TEXT REPRESENTATIONS

Comparison	Shapiro-Wilk p-value	p-value (t-test)	Wilcoxon p-value	Cohen's d	Significance
TF-IDF vs FastText	0,3782	0,0224	0,0125	-0,6625	Significant
TF-IDF vs IndoBERT	0,4603	0,0001	0,0001	-1,3644	Significant
FastText vs IndoBERT	0,0839	0,0008	0,0020	-1,0962	Significant

Prior to conducting the paired t-test, the normality of the paired differences in F1-score was assessed using the Shapiro-Wilk test across 15 paired observations (3 classifiers × 5 folds) for each comparison. The results indicated that the normality assumption was satisfied for all three comparisons ($p = 0.378$ for TF-IDF vs. FastText, $p = 0.460$ for TF-IDF vs. IndoBERT, and $p = 0.084$ for FastText vs. IndoBERT), justifying the use of the parametric paired t-test. As a robustness check, the non-parametric Wilcoxon signed-rank test was also performed on the same paired observations and yielded conclusions fully consistent with the paired t-test results for all comparisons, reinforcing confidence in the statistical significance of the observed performance differences among text representations

Based on Table V, the paired t-test results showed a significant difference between TF-IDF and FastText, with a p-value of 0.0224. The difference between TF-IDF and IndoBERT was also significant, with a p-value of 0.0001, while the difference between FastText and IndoBERT was significant, with a p-value of 0.0008.

The Cohen's d values indicate that the difference between TF-IDF and FastText falls into the medium effect category, while the differences between TF-IDF and IndoBERT and between FastText and IndoBERT indicate larger effects. The negative sign in the Cohen's d values indicates that the second representation in each comparison pair had a higher mean F1-score. Therefore, both statistically and practically, IndoBERT demonstrated superior performance compared to the other two representations.

However, the results of this statistical test should still be interpreted cautiously because the paired observations were derived from five-fold cross-validation across three classifiers. Therefore, the statistical test in this study was used as supporting evidence for performance differences, rather than as the sole basis for drawing conclusions.

E. Error Analysis

Error analysis was conducted to identify the types of errors made by each model after the hyperparameter tuning process. Model errors were categorized into false positives and false negatives. A false positive occurs when a negative review is predicted as positive, while a false negative occurs when a positive review is predicted as negative.

The results show that the combination of IndoBERT with Logistic Regression produced the lowest error rate, at 0,0735 or 7,35%, with 294 misclassifications out of a total of 4.000 reviews. This configuration produced 128 false positives and 166 false negatives. The next lowest error rate was achieved by IndoBERT with Random Forest at 8,33%, followed by IndoBERT with SVM at 9,55%. These findings reinforce the previous evaluation results, indicating that IndoBERT is the most effective representation for sentiment classification of travel application reviews. The summary of model error rates after tuning is presented in Table VI.

TABLE VI
SUMMARY OF MODEL ERROR RATES AFTER TUNING

Model	Representation	Total Samples	Misclassifications	Error Rate	False Positive	False Negative
Logistic Regression	IndoBERT	4.000	294	0,0735	128	166
Random Forest		4.000	333	0,0833	94	239
SVM		4.000	382	0,0955	189	193
Logistic Regression	FastText	4.000	419	0,1047	146	273
Random Forest		4.000	488	0,1220	115	373
SVM		4.000	425	0,1062	141	284
Logistic Regression	TF-IDF	4.000	427	0,1067	192	235
Random Forest		4.000	505	0,1263	250	255
SVM		4.000	427	0,1067	181	246

Based on the patterns of false positives and false negatives, FastText tended to produce a higher number of false negatives than false positives. For example, Random Forest with FastText produced 373 false negatives and only 115 false positives. This pattern indicates that FastText-based models more frequently predicted positive reviews as

negative. This may occur because FastText document representations were formed by averaging word vectors, causing implicit positive words or phrases to lose their influence in the final representation.

In TF-IDF, the number of false positives and false negatives was relatively more balanced, especially in Random Forest with TF-IDF, which produced 250 false positives and 255 false negatives. This indicates that frequency-based representations are able to capture explicit words in both classes, although they still struggle when sentiment appears in implicit, ambiguous, or negation-based forms.

IndoBERT showed a better error pattern, particularly when combined with Logistic Regression. The combination of IndoBERT and Logistic Regression produced 128 false positives and 166 false negatives. These numbers were lower than almost all other configurations. This indicates that IndoBERT's contextual representation was better able to capture variations in sentiment expression in both positive and negative reviews. The error rate comparison across tuned models is visualized in Figure 4.

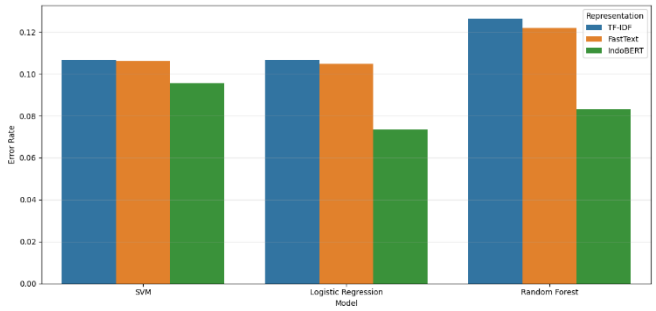

Figure 4. Error rate of tuned models

Several error patterns were repeatedly found across models. In false negative cases, the models often failed to identify positive reviews that were factual and did not contain explicit sentiment words. An example is the review: "Proses Refund kurang lebih 6 jam setelah proses pengajuan. Refund langsung ke nomor rekening jumlah full 100%". This review reflects a positive experience because the refund process was successful. However, because it was expressed informatively without words such as "bagus", "puas", or "cepat", the model could classify it as negative.

False negatives also appeared in reviews containing implicit sentiment or indirect context. An example is the review: "kenapa pas saya cari tiket pesawat yang muncul hanya dari maskapai merah ya? saya juga butuh penawaran dari maskapai lain supaya lebih bebas memilih". This review contains a user need or expectation regarding service variety, but its sentence structure does not explicitly indicate positive or negative polarity, making it difficult for the model to classify accurately.

Meanwhile, false positives often appeared in negative reviews that contained positive words within a negation structure. An example is "gak bagus jangan dipakai ni aplikasi". In this case, the word "bagus" may trigger a positive prediction, especially in word-frequency-based models, even though the overall sentence context is actually negative due to the negation "gak" and the prohibition "jangan dipakai". This type of error highlights the importance of representations that can capture context and word relationships. Previous studies have also shown that bag-of-ngrams-based approaches have limitations in handling contrastive sentences, informal language, and negation structures [8], [31].

Overall, the error analysis shows that IndoBERT was more effective in reducing the number of classification errors, especially when combined with Logistic Regression. However, all models still faced challenges in reviews containing negation, implicit sentiment, very short sentences, and reviews that combined factual information with opinions.

F. Feature Importance

Feature importance analysis was conducted to understand the features or words that contributed most to the sentiment classification process. In the TF-IDF representation, interpretation can be performed directly through words or phrases with the highest weights in each class. This analysis was mainly conducted on the SVM, Logistic Regression, and Random Forest models after tuning.

In the SVM model with TF-IDF, features contributing to positive sentiment included "mudah", "mantap", "bagus", "cepat", "ok", "murah", "keren", "oke", "lumayan", "terima", "puas", "gampang", and "best". Meanwhile, features contributing to negative sentiment included "kecewa", "update", "eror", "mahal", "bayar", "iklan", "buka", "tipu", "jelek", "refund", "buruk", "ganggu", "paylater", "sulit", and "susah".

In the Logistic Regression model, the feature patterns were relatively consistent with SVM. The dominant positive features included "mudah", "mantap", "bagus", "ok", "cepat", "oke", "keren", "murah", "lumayan", "terima", and "puas". The dominant negative features included "kecewa", "update", "eror", "bayar", "refund", "iklan", "mahal", "tipu", "ganggu", "buruk", "jelek", "paylater", "sistem", and "susah". This consistency indicates that these words had strong and stable contributions in distinguishing positive and negative sentiment in travel application reviews. Examples of dominant TF-IDF features in the linear models are presented in Table VII.

TABLE VII
EXAMPLES OF DOMINANT TF-IDF FEATURES IN LINEAR MODELS

Dominant Positive Features	Dominant Negative Features
mudah	*kecewa*
mantap	*update*
bagus	*eror*
cepat	*bayar*
ok / oke	*refund*
murah	*iklan*
puas	*mahal*
best	*tipu*
gampang	*susah*
terima	*paylater*

The positive words that appeared were generally intuitive in the context of travel application reviews. Words such as “mudah”, “cepat”, “murah”, “mantap”, and “bagus” describe user satisfaction with ease of application use, transaction speed, affordable prices, and service quality that meets expectations. Conversely, negative words such as “refund”, “bayar”, “eror”, “mahal”, “kecewa”, “susah”, and “ganggu” indicate complaints related to transactions, refunds, technical problems, prices, and service quality.

These results show that TF-IDF has an advantage in terms of interpretability. Words with high weights can be directly associated with service issues in the travel domain, such as payment, refunds, tickets, hotels, paylater, and application errors. Therefore, although TF-IDF did not outperform IndoBERT, it still has practical value because it can help explain which aspects influence user sentiment.

In Random Forest with TF-IDF, feature interpretation needs to be performed more cautiously. Unlike linear models, Random Forest does not produce the direction of positive or negative contribution through coefficients. Feature importance values in Random Forest indicate how much a feature is used in the splitting process of decision trees, not whether the feature pushes the prediction toward the positive or negative class. Therefore, the interpretation of TF-IDF features in SVM and Logistic Regression is clearer than in Random Forest.

For FastText and IndoBERT, interpretability analysis was performed using permutation importance on embedding dimensions. However, these results have limitations because embedding dimensions cannot be directly translated into specific words. For example, in FastText and IndoBERT, important features are represented as numerical dimensions such as “Dim 95” or “Dim 275”, rather than as words or phrases. Therefore, permutation importance in FastText and IndoBERT was used as a complement to examine the contribution of feature dimensions, not as the main linguistic interpretation.

In general, the feature importance results show that TF-IDF is superior in terms of interpretability, while IndoBERT is superior in predictive performance. Thus, the choice of text representation should consider the intended use. If the main objective is accuracy and the ability to capture context, IndoBERT is more recommended. However, if the main objective is interpretability and ease of explanation, TF-IDF remains a strong option.

G. Discussion

The results of this study show that the IndoBERT representation generally achieved the best performance compared to TF-IDF and FastText. Under the default setting, the combination of IndoBERT with Random Forest produced the highest F1-score of 0,9152. After hyperparameter tuning, the best combination changed to IndoBERT with Logistic Regression, with an F1-score of 0,9261. This finding indicates that IndoBERT’s contextual representation is effective for sentiment classification of Indonesian-language travel application reviews.

The advantage of IndoBERT can be explained by its ability to generate representations that consider the overall sentence context. In user reviews, sentiment does not always appear through explicit words such as “bagus” or “buruk”, but may also appear through sentence structure, negation, indirect complaints, or factual experiences. With contextual representation, IndoBERT is better able to capture relationships between words compared to frequency-based TF-IDF or FastText, which uses static embeddings. This is consistent with BERT, which was designed to build bidirectional text representations, and IndoBERT, which was developed specifically for Indonesian NLP tasks [15], [16].

Nevertheless, TF-IDF still showed competitive performance. After tuning, TF-IDF with Logistic Regression achieved an F1-score of 0,8927, while TF-IDF with SVM achieved an F1-score of 0,8926. These results indicate that in relatively short travel application reviews, explicit words remain highly informative for sentiment classification. Words such as “mudah”, “bagus”, “cepat”, “refund”, “eror”, “kecewa”, and “susah” can serve as strong indicators in distinguishing positive and negative reviews. Therefore, TF-IDF remains relevant as a lightweight, efficient, and interpretable alternative. This finding is in line with studies on Traveloka reviews showing that TF-IDF and SVM can still serve as strong baselines when hyperparameter optimization is performed systematically [8].

FastText also produced fairly good performance but was unable to outperform IndoBERT. After tuning, FastText with Logistic Regression achieved an F1-score of 0,8943, and FastText with SVM achieved an F1-score of 0,8928. This performance indicates that FastText’s subword information helps handle word variations, but it is not always sufficient to capture full sentence context. One possible reason is that document representations were formed by averaging word vectors, which may reduce information about word order, negation, and relationships between words. Thus, the theoretical advantage of FastText in handling rare words and morphological variations does not automatically result in the highest performance on every dataset.

The consistently strong performance of Logistic Regression is also an important finding. After tuning, Logistic Regression became the best classifier when combined with IndoBERT. This indicates that high-quality feature representations can make the data easier to separate linearly. In other words, classifier complexity is not always the main factor; the quality of text representation plays a greater role. Logistic Regression is also more efficient in handling high-dimensional features such as 768-dimensional IndoBERT vectors.

Random Forest achieved the best performance under the default setting when combined with IndoBERT, but after tuning, its performance remained below Logistic Regression. This may occur because Random Forest can capture non-linear patterns but is less efficient in high-dimensional feature spaces. In representations such as IndoBERT, the generated features are already sufficiently informative, allowing linear

models such as Logistic Regression to utilize them more effectively.

From the error analysis perspective, IndoBERT with Logistic Regression had the lowest error rate, at 7,35%. This shows that this combination was not only superior based on F1-score but was also able to reduce the number of classification errors. However, several types of reviews were still difficult to classify, particularly factual reviews without explicit sentiment words, reviews containing negation, and reviews containing a mixture of positive and negative information. This finding is consistent with research on tourism user-generated content, which shows that models still face challenges when dealing with reviews containing multiple aspects or mixed sentiment [18].

Practically, the findings of this study provide two implementation alternatives. If computational resources are available and the main goal is to obtain the best performance, IndoBERT with Logistic Regression is recommended. However, if the system requires a lightweight, fast, and easily explainable model, TF-IDF with SVM or Logistic Regression can be a competitive alternative. TF-IDF is also easier to interpret because the features are represented as words or phrases that can be directly associated with service issues.

This study has several limitations. First, the dataset was derived from only two travel applications, namely Traveloka and Tiket.com, so the findings may not be directly generalizable to other application domains. Second, classification was performed only on two sentiment classes, namely positive and negative, while neutral reviews were not analyzed. Third, IndoBERT was used as a frozen feature extractor, not through end-to-end fine-tuning. Fourth, the statistical testing was conducted based on five folds, so the results should be interpreted cautiously.

Future research is suggested to expand the dataset to more applications and digital service domains, add a neutral class, and perform end-to-end IndoBERT fine-tuning. In addition, aspect-based sentiment analysis can be applied to identify specific aspects that contribute to user satisfaction or complaints, such as refunds, prices, promotions, payments, customer service, tickets, hotels, and technical issues in the application.

IV. CONCLUSION

This study compared TF-IDF, FastText, and IndoBERT representations combined with SVM, Logistic Regression, and Random Forest for sentiment classification of Indonesian travel application reviews. The results show that IndoBERT achieved the best overall performance, particularly when combined with Logistic Regression after hyperparameter tuning, with an F1-score of 0,9261. Its superiority is attributed to its ability to capture contextual meaning, negation, implicit sentiment, and linguistic variations in user reviews. TF-IDF remained competitive after tuning, achieving F1-scores of 0,8927 with Logistic Regression and 0,8926 with SVM. This indicates that frequency-based representations are still effective for short review texts containing explicit sentiment indicators such as “mudah”, “bagus”, “refund”, and “kecewa”. FastText achieved its best F1-score of 0,8943, but its performance did not surpass IndoBERT, likely because averaging word vectors limits its ability to preserve word order and sentence context.

The paired t-test results showed statistically significant differences among all text representations, with Cohen’s d indicating a medium effect for TF-IDF versus FastText and large effects for comparisons involving IndoBERT. Error analysis further showed that IndoBERT with Logistic Regression produced the lowest error rate of 7,35%, although all models still struggled with factual reviews, negation, and implicit sentiment.

Overall, the findings indicate that text representation quality has a greater influence on sentiment classification performance than classifier complexity. For practical implementation, IndoBERT with Logistic Regression is recommended when computational resources are available, while TF-IDF with SVM or Logistic Regression can serve as efficient and interpretable alternatives. Future research should expand the dataset, include more platforms and domains, perform end-to-end IndoBERT fine-tuning, and explore aspect-based sentiment analysis.

ACKNOWLEDGMENTS

The authors express their gratitude to IPB University for its institutional support, research facilities, and academic environment that enabled the completion of this study. In particular, the authors would like to extend their sincere appreciation to the School of Data Science, Mathematics, and Informatics, IPB University, for providing essential academic guidance, research resources, and a highly supportive learning atmosphere throughout the research process.

REFERENCES

[1] W. Chen, Z. Xu, X. Zheng, Q. Yu, and Y. Luo, “applied sciences Research on Sentiment Classification of Online Travel Review Text,” *Appl. Sci.*, vol. 10, 2020, doi: 10.3390/app10155275.

[2] Y. A. Singgalen, “Performance Analysis of IndoBERT for Sentiment Classification in Indonesian Hotel Review Data,” *J. Inf. Syst. Res.*, vol. 6, no. 2, pp. 976–986, 2025, doi: 10.47065/josh.v6i2.6505.

[3] M. G. Al Hakim and F. Irwiensyah, “Analisis Sentimen Terhadap Ulasan Pengguna Pada Aplikasi Traveloka Menggunakan Metode Naïve,” *Build. Informatics, Technol. Sci.*, vol. 6, no. 3, pp. 1448–1456, 2024, doi: 10.47065/bits.v6i3.6119.

[4] H. Jayadianti, W. Kaswidjanti, A. T. Utomo, S. Saifullah, F. A. Dwiyanto, and R. Drezewski, “Sentiment analysis of Indonesian reviews using fine-tuning IndoBERT and R-CNN,” *Ilk. J. Ilm.*, vol. 14, no. 3, pp. 348–354, 2022, doi: 10.33096/ilkom.v14i3.1505.348-354.

[5] A. Muhammad, S. Defit, and G. W. Nurcahyo, “Determining Intent : Sentiment Analysis Based on the Classification of Indonesian Tourist Destination Review Texts,” *J. Adv. Inf. Technol.*, vol. 15, no. 10, 2024, doi: 10.12720/jait.15.10.1106-1116.

[6] S. Suryadi, D. Syahputra, N. Astrianda, R. A. Syahputra, and R. Suhendra, “Leveraging Machine Learning for Sentiment Analysis in Hotel Applications: A Comparative Study of Support Vector

Machine and Random Forest Algorithms," *Brill. Res. Artif. Intell.*, vol. 4, no. 2, pp. 567–576, 2024, doi: 10.47709/brilliance.v4i2.4877.

[7] R. K. Mishra, S. Urolagin, and A. A. J. Jothi, "A Sentiment analysis-based hotel recommendation using TF-IDF Approach," *Proc. 2019 Int. Conf. Comput. Intell. Knowl. Econ. ICCIKE 2019*, pp. 811–815, 2019, doi: 10.1109/ICCIKE47802.2019.9004385.

[8] M. B. Kurniawan, R. Hikmianto, and I. Muslihah, "Hyperparameter Optimization of TF-IDF and SVM via Grid Search for Sentiment Analysis of Traveloka Customer Reviews," *Khazanah Inform.*, vol. 11, no. 2, 2025.

[9] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information," *Assoc. Comput. Linguist.*, vol. 5, pp. 135–146, 2017.

[10] N. Khamphakdee and P. Seresangtakul, "An Efficient Deep Learning for Thai Sentiment Analysis," *Data*, vol. 8, no. 90, 2023, doi: doi.org/10.3390/data8050090.

[11] A. M. M. Al Zoubi, *Spam Reviews Detection Models in Multilingual Contexts applying Sentiment Analysis , Metaheuristics , and Advanced Word Embedding*. Spain: Universidad De Granada, 2024.

[12] H. Suroyo and E. J. Pratama, "Comparison of Text Representation Methods for Sentiment Analysis Using Support Vector Machine," *J. Adv. Inf. Ind. Technol.*, vol. 7, no. 1, pp. 21–30, 2025, doi: 10.52435/jaiit.v7i1.610.

[13] L. Afuan and N. Hidayat, "Sentiment Analysis of the Kampus Merdeka Program on Twitter Using Support Vector Machine," *J. Appl. Data Sci.*, vol. 5, no. 4, pp. 1738–1753, 2024.

[14] F. I. Ramadhani, T. A. Yoga, and N. A. Verdhika, "Komparasi FastText dan TF-IDF Berbasis Random Forest pada Analisis Sentimen IKN di Youtube," vol. 6, no. 12, pp. 2288–2301, 2026, doi: 10.47065/tin.v6i12.9749.

[15] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *NAACL HLT 2019 - 2019 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf.*, vol. 1, no. Mlm, pp. 4171–4186, 2019.

[16] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP," *COLING 2020 - 28th Int. Conf. Comput. Linguist. Proc. Conf.*, pp. 757–770, 2020, doi: 10.18653/v1/2020.coling-main.66.

[17] M. A. K. Fata, S. Sumpeno, A. D. Wibawa, and D. A. Feryando, "Evaluating the Sentiment Analysis from Auto-Generated Summary Text Using IndoBERT Fine-Tuning Model in Indonesian News Text," *Proc. - 2023 15th IEEE Int. Conf. Comput. Intell. Commun. Networks, CICN 2023*, pp. 822–829, 2023, doi: 10.1109/CICN59264.2023.10402345.

[18] R. I. Perwira, V. A. Permadi, D. I. Purnamasari, and R. P. Agusdin, "Domain-Specific Fine-Tuning of IndoBERT for Aspect-Based Sentiment Analysis in Indonesian Travel User-Generated Content," *J. Inf. Syst. Eng. Bus. Intell.*, vol. 11, no. 1, pp. 30–40, 2025, doi: 10.20473/jisebi.11.1.30-40.

[19] A. Hogenboom, D. Bal, F. Fransincar, M. Bal, F. de Jong, and U. Kaymak, "Exploiting Emoticons in Sentiment Analysis," *Proc. 28th Annu. ACM Symp. Appl. Comput.*, 2013.

[20] A. Deshmukh, A. Dhage, R. Gadapa, S. Butle, A. Yenkikar, and N. P. Sable, "Comparative Analysis of Machine Learning Algorithms for Emotion Classification," *2024 IEEE Pune Sect. Int. Conf. PuneCon 2024*, pp. 1–6, 2024, doi: 10.1109/PuneCon63413.2024.10895276.

[21] T. A. Almeida, J. M. G. Hidalgo, and A. Yamakami, "Contributions to the Study of SMS Spam Filtering : New Collection and Results," *Proc. 11th ACM Symp. Doc. Eng.*, 2011.

[22] J. Ramos, "Using TF-IDF to determine word relevance in document queries," Jan. 2003, [Online]. Available: https://api.semanticscholar.org/CorpusID:14638345

[23] C. Cortes and V. Vapnik, "Support-Vector Networks," *Kluwer Acad. Publ.*, vol. 20, pp. 273–297, 1995.

[24] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2013.

[25] P. Awasthi, M. Thomas, D. Junghare, and M. Bianco, "A Machine Learning Framework for Failure Mode Identification from Warranty Data," *Proc. - Annu. Reliab. Maintainab. Symp.*, pp. 1–6, 2025, doi: 10.1109/RAMS48127.2025.10935108.

[26] H. D. Vu, Q. T. Pham, V. K. Solanki, T. M. Hoang, and D. T. Tran, "Sentiment Analysis using Machine Learning and Deep Learning Models," *Proc. - 2024 IEEE Int. Conf. Mach. Learn. Appl. Netw. Technol. ICMLANT 2024*, pp. 68–73, 2024, doi: 10.1109/ICMLANT63295.2024.00017.

[27] L. Breiman, "Random Forests," *Kluwer Acad. Publ.*, vol. 45, pp. 5–32, 2001.

[28] H. Om and A. Kumar Sharma, "Demystifying Existing Sentiment Analysis Approaches of Hindi and English Languages using Machine Learning," *Proc. - IEEE 2024 1st Int. Conf. Adv. Comput. Commun. Networking, ICAC2N 2024*, pp. 1210–1217, 2024, doi: 10.1109/ICAC2N63387.2024.10895523.

[29] S. Srivastava, N. Bala, A. Gupta, B. D. Priya, S. Kumar, and A. Raj, "Optimization of Sentiment Analysis Models Using Bayesian Hyperparameter Tuning," *2024 Int. Conf. Artif. Intell. Quantum Comput. Sens. Appl. ICAIQSA 2024 - Proc.*, pp. 1–6, 2024, doi: 10.1109/ICAIQSA64000.2024.10882347.

[30] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, 2009, doi: 10.1016/j.ipm.2009.03.002.

[31] B. Das and S. Chakraborty, "An Improved Text Sentiment Classification Model Using TF-IDF and Next Word Negation," 2018, [Online]. Available: http://arxiv.org/abs/1806.06407