

Comment Intention Classification in TikTok Fashion Affiliate Marketing Based on Attention-Weighted Word2Vec Representation

Rahmawati Septiani¹⁾*, Masy Ari Ulinuha²⁾, Nur Cahyo Hendro W³⁾

^{1,2,3)}Department of Information Technology, Faculty of Science and Technology, Universitas Islam Negeri Walisongo Semarang, Semarang, Indonesia

¹⁾23080960020@student.walisongo.co.id, ²⁾ulinuha@walisongo.ac.id, ³⁾nur_cahyo@walisongo.ac.id

Submitted : Sep 11, 2026 | Accepted : Oct 15, 2026 | Published : Oct 18, 2026

Abstract: TikTok Fashion Affiliate generates short, informal user comments with a variety of communicative intentions, necessitating a text representation capable of preserving key information within the comments. This study aims to evaluate Norm-Based Softmax Weighting as a non-learned weighting mechanism and compare it with Mean Pooling for classifying the intent of Indonesian-language TikTok Fashion Affiliate comments. The comment dataset was independently annotated by three annotators with a Krippendorff's Alpha of 0.7950 and classified into three classes: Question, Request, and Opinion. Word2Vec Skip-gram with a dimension of 100 was used to generate comment representations, which were then compared using Mean Pooling and Norm-Based Softmax Weighting in Support Vector Machine (SVM) and Logistic Regression models. Hyperparameters were selected via Grid Search using Macro F1 as the optimization criterion. To prevent data leakage, Word2Vec was trained only on the training data and retrained on each fold. Evaluation was performed using the test data, 5-Fold Stratified Cross-Validation, and Leave-One-Video-Out. The results showed that SVM with Norm-Based Softmax Weighting achieved a Macro F1 score of 0.7995, slightly higher than Mean Pooling's 0.7913. In contrast, for Logistic Regression, Mean Pooling achieved a Macro F1 score of 0.7846, which was higher than the 0.7768 achieved by Norm-Based Softmax Weighting. However, ablation analysis and statistical testing did not reveal any significant differences. Leave-One-Video-Out validation revealed performance variations across videos and a decline in performance compared to random data splitting. Thus, Norm-Based Softmax Weighting is a competitive non-learned weighting alternative, but its effectiveness depends on the classifier and data characteristics.

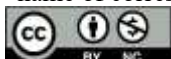
Keywords: Attention, Classification, Pooling, SVM, TikTok, Word2Vec

INTRODUCTION

The growth of social media-based commerce has transformed the way consumers interact with products and content creators. TikTok has become one of the platforms driving this trend through affiliate marketing, where content creators promote products and direct users to make purchases via the platform's shopping features. Fashion affiliate content generates many short, informal comments, such as questions about products, requests for additional information, and opinions on the promoted products. Classifying comments based on communication intent can provide a more specific understanding of user interactions compared to conventional sentiment classification because the focus is on identifying the user's communicative intent, rather than determining whether the user's attitude is positive or negative (Harahap & Sazali, 2024).

Word2Vec has been widely used to represent Indonesian-language text because its vector representations are capable of capturing relationships between words and can be combined with conventional machine learning algorithms (Andriyani et al., 2024). However, the conversion of Word2Vec vectors at the token level into a single representation of a comment generally uses Mean Pooling, in which all tokens in the comment contribute equally to the final vector. Aggregation that assigns equal weights to all tokens may not be able to sufficiently distinguish the contributions of different tokens in representing communicative intent, especially in short social media

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

comments (Hendrawan et al., 2022). This situation highlights the need to test aggregation mechanisms that can assign different weights to each token without unduly increasing the complexity of the representation.

This study uses Attention-Weighted Word2Vec, an approach that combines Word2Vec embeddings with token weighting before aggregating them into a comment representation. Weighting is applied using Norm-Based Softmax Weighting as a non-learned weighting mechanism. This mechanism calculates the L2 norm of each Word2Vec token embedding, then normalizes the score using the softmax function to generate token weights. Unlike learned attention in Transformer-based architectures, which learns parameters through the training process, this approach does not use queries, keys, and values, nor does it use additional attention parameters. The L2 norm is used as a weighting heuristic, not as a direct measure of a token's semantic importance. Thus, the effectiveness of the weighting must be evaluated through empirical testing of classification performance.

To address this gap, this study conducted an ablation experiment that explicitly compared Mean Pooling and Norm-Based Softmax Weighting using the same Word2Vec representations. Both aggregation strategies were used to form comment representations, which were then classified into three categories Question, Request, and Opinion using Support Vector Machines (SVM) and Logistic Regression. A comparison of the two classifiers was conducted to determine the consistency of the aggregation strategies' impact on classification results. To enhance methodological reliability, Word2Vec was trained exclusively on the training data and retrained separately on each fold during the representation generation process. Hyperparameter selection for classification was performed using a grid search with Macro F1 as the optimization criterion. Additionally, the study evaluated the reliability of the annotations and tested the model's generalization ability through validation on videos not used in the training process.

Based on these issues, this study aims to evaluate whether Attention-Weighted Word2Vec using Norm-Based Softmax Weighting produces representations that are competitive compared to conventional Mean Pooling in the classification of the intent of Indonesian-language TikTok Fashion Affiliate comments. The contributions of this study encompass three aspects: providing an independently annotated intent classification dataset; evaluating a lightweight, untrained weighting mechanism against Mean Pooling as a baseline; and testing the model's reliability and generalization ability through cross-validation with data leakage control and video-based evaluation.

LITERATURE REVIEW

Word2Vec is a word embedding method that represents words as vectors of a certain dimension based on their relationships with the surrounding context. This representation allows words with similar usage patterns to occupy nearby positions in vector space. In Indonesian text processing, Word2Vec has been used for various text classification tasks, including sentiment analysis of product reviews (Andriyani et al., 2024), user sentiment toward investment apps and platforms (Suharman et al., 2025), and fake news detection (Asramanggala et al., 2023). In Word2vec-based research, multiple tokens need to be combined into a single vector that represents the entire text. One commonly used approach is mean pooling, which involves calculating the average of the vectors of all tokens. This approach is simple and requires no additional parameters, but it assigns equal weight to each token and thus does not account for differences in how tokens contribute to the representation of comments (Hendrawan et al., 2022).

Attention-Weighted Word2Vec is an approach that combines Word2Vec representations with token weighting before the vectors are aggregated into a comment representation. This approach allows each token to contribute differently to the final representation, unlike mean pooling, which assigns equal weight to all tokens. Embedding-based weighting has been used in several text representation approaches to enhance the information retained during the aggregation process (Zhao et al., 2022). In this study, this concept is implemented through Norm-Based Softmax Weighting, a norm-based weighting strategy applied to Word2Vec vectors to generate comment representations. This approach is positioned as a non-learned aggregation mechanism and is evaluated through a comparison with mean pooling on the comment intent classification task.

Support Vector Machines (SVMs) are one of the most widely used classification methods for text data because they can handle high-dimensional feature representations. The combination of SVMs and Word2Vec has been applied to various text classification tasks and has demonstrated competitive performance (Rizky & Lhaksana, 2023). SVM performance is also influenced by hyperparameter selection, particularly when using the Radial Basis Function (RBF) kernel; therefore, hyperparameter tuning is necessary to obtain an appropriate model configuration (Wicaksana & Huda, 2025).

Logistic regression was used as a benchmark classifier against SVM because it is a relatively simple and efficient classification method for text data (Saputra et al., 2025). Using both classifiers allows us to test whether the effects of Attention-Weighted Word2Vec representations and mean pooling remain evident across different classification algorithms. Thus, the evaluation focuses not only on differences in aggregation strategies but also on the consistency of their performance when used with different classifiers.

Intent classification is an approach for identifying the underlying goals or intentions of an utterance based on linguistic information contained in the text. In the analysis of social media comments, this approach can be used

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

to distinguish the communicative functions conveyed by users, so that the information obtained is not limited to sentiment polarity. Research on comments on digital platforms shows that modeling user intent can be used to identify different forms of interaction based on their communicative purposes (Harahap & Sazali, 2024). In this study, this concept is applied to TikTok Fashion Affiliate comments, focusing on “Question,” “Request,” and “Opinion” as categories of intent. These categories serve as the basis for evaluating the ability of Word2Vec-based representations to distinguish users’ communicative intentions.

METHOD

Research Framework

This study uses the Attention-Weighted Word2Vec framework to classify the intent of comments on TikTok Fashion Affiliate content. The research framework applies Norm-Based Softmax Weighting as a non-learned weighting mechanism to assign weights to Word2Vec token representations before classification using Support Vector Machines (SVM) and Logistic Regression. To evaluate the contribution of this weighting mechanism, the study compares Norm-Based Softmax Weighting with Mean Pooling using the same Word2Vec representations.

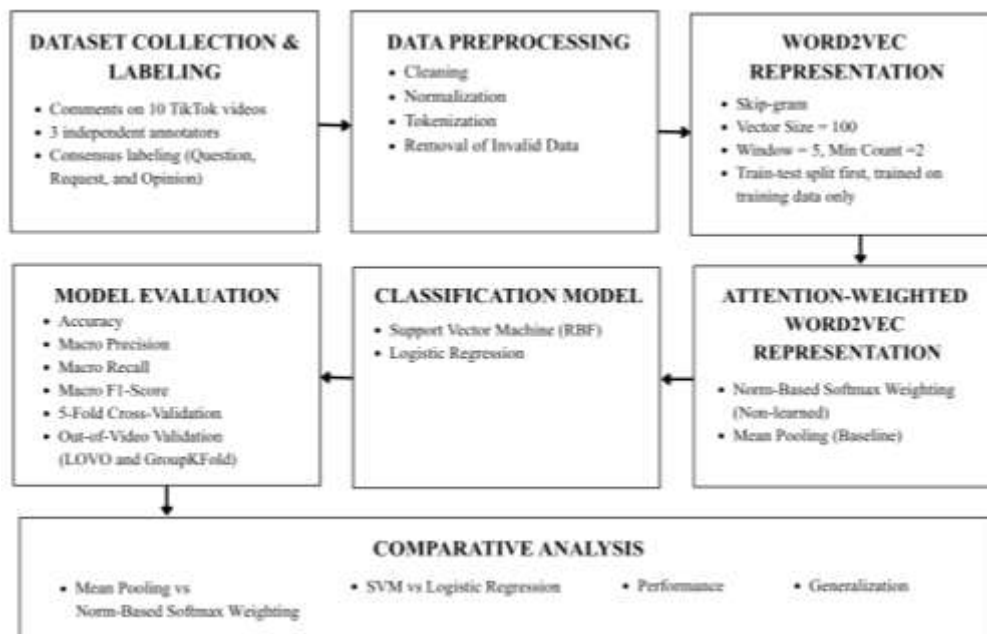


Figure 1 Research Framework

Figure 1 shows the research workflow, which consists of several main stages, starting from the collection and labeling of comments through to the model classification and evaluation process. The comment data obtained from TikTok Fashion Affiliate video content first undergoes a preprocessing stage to produce more structured text. Next, the text is represented using Word2Vec with the Skip-gram approach, then aggregated using two strategies: Mean Pooling as the baseline and Norm-Based Softmax Weighting as a non-learned weighting mechanism. Both representations are used as input for SVM and Logistic Regression, with hyperparameter selection performed via Grid Search. Evaluation is conducted using classification metrics, 5-Fold Stratified Cross-Validation, and video-based validation using LOVO and GroupKFold to compare the performance, stability, and generalization ability of each combination of representation and classifier.

Data Collection and Labeling

The data for this study was obtained by scraping user comments from 10 TikTok videos classified as “Fashion Affiliate” content. The videos were selected based on their engagement levels, number of comments, and relevance to the promotion of women’s fashion products. The scraping process yielded 4,215 comments as the initial dataset for the study.

The collected comments were then manually labeled into three intent categories: Question, Request, and Opinion. “Question” includes product-related inquiries; “Request” includes requests for information or specific actions; and “Opinion” includes statements or evaluations of products or content.

Labeling was performed by three independent annotators: a researcher, a TikTok user, and an affiliate content creator. Each annotator read the comments in their entirety and determined the category based on context and

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

communicative intent, with the most dominant intent used if the comment contained more than one form of communication. Disagreements over labels were resolved through majority voting; the label receiving at least two out of three votes was designated as the consensus label. Inter-annotator agreement was evaluated using Cohen's Kappa for each pair of annotators, as well as Fleiss' Kappa and Krippendorff's Alpha for overall agreement. Agreement calculations were performed on 4,215 initial comments prior to the preprocessing stage. The consensus label was used as the ground truth in the creation of the classification dataset.

Data Preprocessing

The text preprocessing stage is performed to clean and normalize TikTok comments before the feature extraction process. This process aims to reduce noise in the text data, standardize token representations, and improve the quality of the input for the embedding generation stage. Given that TikTok comments tend to be short, informal, and contain a wide variety of slang, preprocessing is a crucial step in producing more consistent text representations.

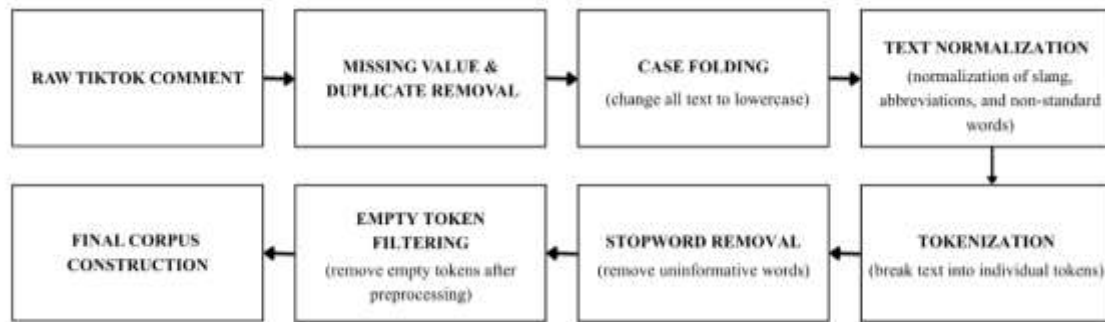


Figure 2 Data Preprocessing

Word2Vec Feature Representations

Text feature representations were created using the Word2Vec technique with a Skip-gram architecture. This approach was used to transform the processed tokens into low-dimensional numerical vectors capable of capturing the semantic relationships between words found in comments related to TikTok Fashion Affiliate (Ayuningtyas & Tantyoko, 2024).

Suppose a comment is represented as a token T . Each token t_i is then mapped to a d -dimensional embedding vector v_i in this study, d is set to 100 dimensions:

$$T = \{t_1, t_2, \dots, t_n\} \quad (1)$$

$$t_i \rightarrow v_i \in \mathbb{R}^d \quad (2)$$

The Word2Vec model was configured with a window size of 5, a minimum word frequency (min-count) of 2, and 100 epochs. The resulting vector representations were then used to form comment representations through two aggregation strategies, Mean Pooling as the baseline and Norm-Based Softmax Weighting as a non-learned weighting mechanism.

To prevent data leakage, Word2Vec training was conducted exclusively using the training data. During the cross-validation process, the Word2Vec model is retrained on the training data in each fold and used to represent the validation data in that fold. Thus, neither the validation data nor the test data is used in the creation of the vocabulary or embeddings. Tokens in the validation or test data that are not present in the training data vocabulary are ignored, and comments that do not contain in-vocabulary tokens are represented by zero vectors.

Attention-Weighted Embedding

Each token represented using Word2Vec must be aggregated into a single vector that represents the entire comment. Mean Pooling is performed by calculating the average of all token vectors, so that each token contributes equally to the final representation (Hara et al., 2026). If a comment consists of n tokens with embedding vectors v_i , the comment representation is calculated using the equation:

$$Z_{mean} = \frac{1}{n} \sum_{i=1}^n v_i \quad (3)$$

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Mean Pooling is used as a baseline to compare performance with the Norm-Based Softmax Weighting method, which involves non-learned weighting based on the L2 norm of vector embeddings. The embedding norm is used as magnitude-based information (Liu et al., 2020). The L2 norm is used as a heuristic to determine relative weights, not as a direct measure of a token's semantic importance.

For each token, the weighting score is calculated based on the L2 norm of the embedding vector:

$$score_i = ||v_i||_2 \quad (4)$$

$$\alpha_i = softmax(score_i) = \frac{exp(score_i - max_k(score_k))}{\sum_{j=1}^n exp(score_j - max_k(score_k)) + \epsilon} \quad (5)$$

The weights obtained are used to generate a representation of the comments through weighted aggregation:

$$z = \sum_{i=1}^n \alpha_i \cdot v_i \quad (6)$$

Unlike Mean Pooling, which assigns equal weights to each token, this mechanism assigns weights based on the L2 norm of the embedding in a non-learned manner. Thus, the attention mechanism used here is norm-based weighting, rather than learned attention as in the Transformer. Its effectiveness is evaluated by comparing it to Mean Pooling using the same classifier.

Classification Model

The feature representations generated by the Mean Pooling and Norm-Based Softmax Weighting aggregation processes are used as input for comment intent classification. This study uses two machine learning classification algorithms: Support Vector Machine (SVM) and Logistic Regression.

Support Vector Machine (SVM)

SVM is used as one of the classifiers to model the decision boundary on the Word2Vec feature representation. The Radial Basis Function (RBF) kernel is used to handle the nonlinear relationships between features in distinguishing the three categories of comment intent. The parameters C and γ (gamma) were optimized via Grid Search, using Macro F1 as the model selection criterion. Class weights were set to “balanced” to mitigate the impact of class imbalance.

Logistic Regression

Logistic Regression is used as a baseline model to evaluate the impact of representation strategies on intent classification. This model is used to predict three intent categories based on feature representations generated by Mean Pooling and Norm-Based Softmax Weighting. The parameter C is optimized using Grid Search with Macro F1 as the selection criterion.

Hyperparameter Tuning and Evaluation

Hyperparameter selection was performed using grid search on the training data with 5-fold cross-validation. Each hyperparameter combination was evaluated using Macro F1 as the scoring function, and the combination with the highest average Macro F1 value was selected as the optimal set of parameters. For the SVM with a Radial Basis Function (RBF) kernel, the grid search was conducted by testing values of C equal to 0.1, 1, 10, and 100, as well as values of γ values of “scale” equal to 0.01, 0.1, and 1. Meanwhile, Logistic Regression is tuned by testing values of C equal to 0.01, 0.1, 1, 10, and 100. The search process is performed separately for each combination of classifier and aggregation strategy, namely Mean Pooling and Norm-Based Softmax Weighting.

In each fold, Word2Vec was trained using the training data and applied to the validation data. The best hyperparameters were selected based on the average Macro F1 score across the five folds. The test data was not used in the tuning process but was used only for final evaluation using Accuracy, Macro Precision, Macro Recall, and Macro F1-score. The same procedure was applied to SVM and Logistic Regression for both representation strategies.

Token Weighting Analysis

The analysis was conducted to evaluate the relationship between the L2 norm and token relevance to a class using mutual information on the training data. Spearman's correlation was calculated for tokens with a minimum document frequency of 5, along with partial correlations with document frequency as a control variable. On the

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

test data, the maximum distributions were observed for uniform weights, normalized entropy, and cosine similarity relative to the Mean Pooling representation in comments containing at least two in-vocabulary tokens.

RESULT

Dataset and Annotation Results

A total of 4,215 comments from 10 TikTok Fashion Affiliate videos were collected as the initial dataset. The annotation results from three annotators were then compared to determine agreement and consensus labels. Examples of annotation results, consensus labels, and annotator agreement scores are presented in Table 1.

Table 1 Annotation Examples and Inter-Annotator Agreement

Comment	A1	A2	A3	Label Consensus	Number of Votes	Agreement Status
kak yang rok tutu pink sama kemejanya beli dimana?	Question	Question	Question	Question	3	Fully Agreement
keranjang kuningnya mana kk 🤔	Question	Question	Request	Question	2	Majority Agreement (2 out of 3)
kak tolong spill semuanya	Request	Request	Request	Request	3	Fully Agreement
bagus bgt sukaaa	Opinion	Opinion	Opinion	Opinion	3	Fully Agreement
link nya mana kka	Question	Question	Request	Question	2	Majority Agreement (2 out of 3)
Agreement Measure						Value
Cohen's Kappa (A1 vs A2)						0.7336
Cohen's Kappa (A1 vs A3)						0.7004
Cohen's Kappa (A2 vs A3)						0.9413
Fleiss' Kappa						0.7949
Krippendorff's Alpha						0.7950

The annotation results show that 3,631 comments (86.1%) received full agreement, while 584 comments (13.9%) had label discrepancies and were resolved through majority voting. For comments with label discrepancies, the category receiving at least two out of three votes was designated as the consensus label. Agreement measurement results show that the annotation process exhibits a fairly strong to very high level of agreement among annotators. The A2 vs. A3 pair demonstrated high consistency, while the A1 vs. A2 and A1 vs. A3 pairs showed good levels of agreement. Overall, Fleiss' Kappa and Krippendorff's Alpha values approaching 0.80 indicate that the three annotators assigned labels relatively consistently. These results demonstrate adequate reliability in the annotation process; therefore, the consensus labels obtained through majority voting were used as the ground truth in the classification stage.

After establishing the consensus labels, the data underwent a preprocessing stage. Removing missing values and duplicates reduced the number of comments from 4,215 to 3,034. Next, 390 comments that did not yield valid tokens after preprocessing were excluded, resulting in 2,644 comments as the final dataset, with an average of 3.8 tokens per comment. The distribution of the final dataset by intent category is presented in Table 2.

Table 2 Final Dataset Distribution

Intention Class	Number of Comments	Percentage
Question	495	18.7%
Request	627	23.7%
Opinion	1,522	57.6%
Total	2,644	100%

Based on Table 2, the "Opinion" category is the class with the highest number of comments, namely 1,522 comments (57.6%), while "Request" and "Question" have 627 comments (23.7%) and 495 comments (18.7%),

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

respectively. This distribution indicates a class imbalance, so evaluation cannot be based solely on accuracy; Macro F1 must be considered to assign equal weight to each class.

The final dataset, consisting of 2,644 comments, was divided into 2,115 training comments (80%) and 529 test comments (20%) using a stratified split. The test data consists of 305 "Opinion" comments, 125 "Request" comments, and 99 "Question" comments. The training data is used for model training and hyperparameter tuning, while the test data is used to evaluate the model's final performance after the hyperparameter tuning process is complete.

Hyperparameter Tuning and Cross-Validation Results

Hyperparameter selection was performed to obtain the best configuration for each combination of classifier and representation strategy. The grid search process was conducted on the training data using 5-fold cross-validation, with Macro F1 as the primary criterion for hyperparameter selection. The results of the best configurations for SVM and logistic regression using mean pooling and norm-based softmax weighting are presented in Table 3.

Table 3 Hyperparameter Tuning

Model	Representation	Best C	Best γ	CV Macro F1
SVM	Mean Pooling	1	1	0.8302
	Norm-Based Softmax	10	scale	0.8153
Logistic Regression	Mean Pooling	10	-	0.7823
	Norm-Based Softmax	100	-	0.7914

Based on Table 3, the SVM with Mean Pooling achieved the highest CV Macro F1 score of 0.8302 with $C = 1$ and $\gamma = 1$. The SVM with Norm-Based Softmax Weighting achieved a CV Macro F1 score of 0.8153 with $C = 10$ and $\gamma = \text{"scale"}$. In Logistic Regression, Norm-Based Softmax Weighting produced a CV Macro F1 score of 0.7914 with $C = 100$, slightly higher than Mean Pooling, which achieved 0.7823 with $C = 10$. Thus, changes in the representation strategy yield different results for the two classifiers: the CV Macro F1 score decreased by 0.0149 for the SVM when using Norm-Based Softmax Weighting, while for Logistic Regression, it increased by 0.0091 compared to Mean Pooling.

After the optimal hyperparameters were obtained, the model was evaluated using 5-fold stratified cross-validation on the training data while maintaining the class proportions in each fold. The evaluation was conducted using Accuracy and Macro F1 to determine the performance for each fold and their averages.

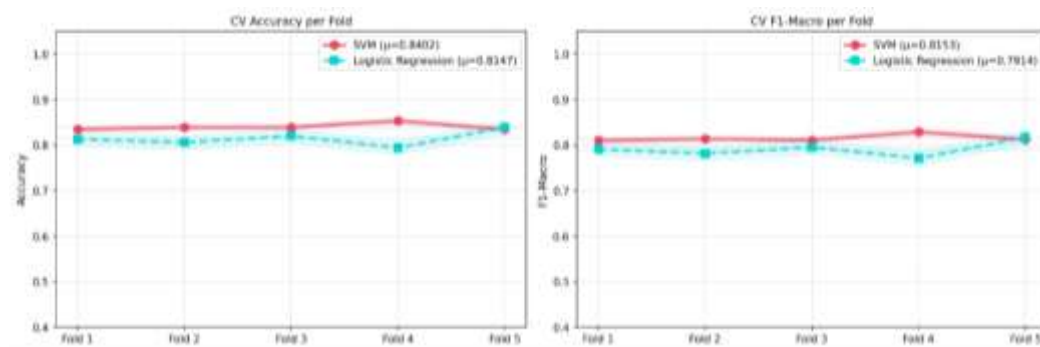


Figure 3 5-Fold Stratified Cross-Validation

Figure 3 shows the results of 5-fold stratified cross-validation based on accuracy and macro F1 score for each fold. In the Norm-Based Softmax Weighting representation, SVM achieved an average Accuracy of 0.8402 and a Macro F1 score of 0.8153, while Logistic Regression achieved an average Accuracy of 0.8147 and a Macro F1 score of 0.7914. These results show variations in performance across the five folds, with the average values used to illustrate the model's performance during the validation process.

Classification Performance Comparison

A comparison of classification performance was conducted to evaluate the effectiveness of two representation strategies Mean Pooling and Norm-Based Softmax Weighting in Support Vector Machine (SVM) and Logistic Regression models for classifying the intent of comments on TikTok Fashion Affiliate content. The evaluation on the test data included a confusion matrix analysis as well as overall metrics, namely Accuracy, Macro Precision,

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Macro Recall, and Macro F1-score. The evaluation results were used to compare the performance of each combination of representation and classifier in classifying the categories Question, Request, and Opinion.

A. Confusion Matrix Analysis

Figure 4 shows the normalized confusion matrix for SVM and Logistic Regression using Norm-Based Softmax Weighting. A comparison with Mean Pooling is conducted using overall performance metrics and ablation analysis. The diagonal values indicate the proportion of predictions that match the true labels, while the off-diagonal values indicate inter-class classification errors.

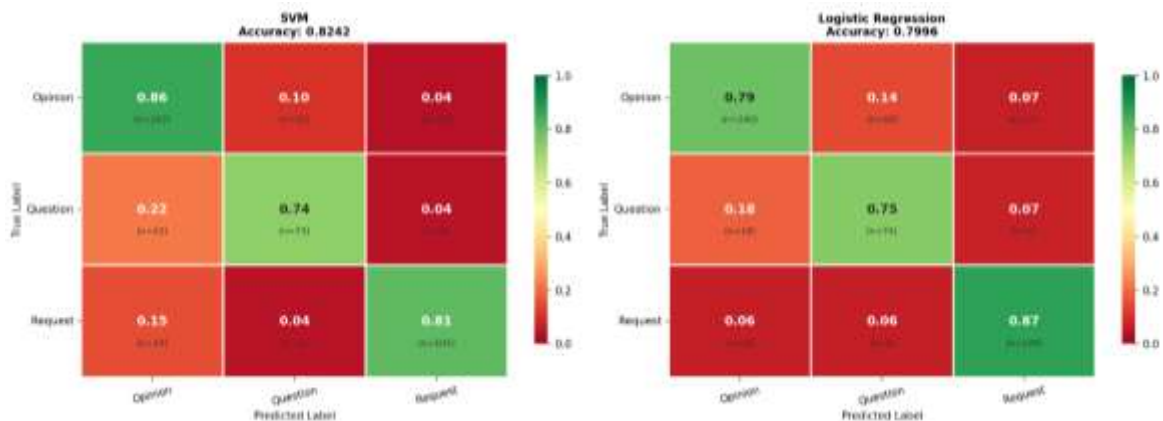


Figure 3 Confusion Matrix of Classification Models

Based on Figure 4, SVM achieved an accuracy of 82.42%, while Logistic Regression achieved 79.96%. For SVM, the proportion of correct predictions for the Opinion, Question, and Request classes was 0.86, 0.74, and 0.81, respectively. For Logistic Regression, the proportions of correct predictions were 0.79, 0.75, and 0.87, respectively. These values indicate that the two models exhibit different prediction patterns across each class, with the highest rate of correct predictions in the “Opinion” class for SVM and in the “Request” class for Logistic Regression.

B. Class-Specific Classification Performance

We compared classification performance across each intent category to assess the ability of each combination of model and representation strategy to distinguish between the Opinion, Question, and Request classes. We evaluated performance using Precision, Recall, and F1-score on the test data. In addition to class-specific performance, we used Accuracy to indicate the overall performance of each combination of model and representation.

Table 4 Classification Performance by Intention Class

Model	Representation	Class	Precision	Recall	F1-Score	Accuracy
SVM (RBF)	Mean Pooling	Opinion	0.8319	0.9246	0.8758	82.99%
		Question	0.7439	0.6162	0.6740	
		Request	0.8889	0.7680	0.8240	
	Norm-Based Softmax Weighting	Opinion	0.8647	0.8590	0.8618	82.42%
		Question	0.6636	0.7374	0.6986	
		Request	0.8707	0.8080	0.8382	
Logistic Regression	Mean Pooling	Opinion	0.9041	0.8033	0.8507	80.91%
		Question	0.6293	0.7374	0.6791	
		Request	0.7746	0.8800	0.8240	
	Norm-Based Softmax Weighting	Opinion	0.9023	0.7869	0.8406	79.96%
		Question	0.5873	0.7475	0.6578	
		Request	0.7956	0.8720	0.8321	

Based on Table 4, the SVM with Mean Pooling achieved an accuracy of 82.99%, while the SVM with Norm-Based Softmax Weighting achieved 82.42%. The highest F1-score for SVM was observed in the Opinion class (0.8758 and 0.8618), while the Question class had the lowest values (0.6740 and 0.6986). For Logistic Regression, the Accuracy was 80.91% and 79.96% for Mean Pooling and Norm-Based Softmax Weighting, respectively. The

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

highest F1-Score was also found in the Opinion class, while the Question class had the lowest scores for both representation methods. These results indicate performance differences across classes and representation strategies, which are further compared using overall metrics and ablation analysis.

C. Overall Performance Evaluation

An overall performance evaluation was conducted to compare the impact of the strategies on SVM and Logistic Regression using the test data. The comparison included Accuracy, Macro Precision, Macro Recall, and Macro F1-score. The evaluation results for the four combinations of models and representations are presented in Table 5.

Table 5 Overall Classification Performance on the Test Set

Model	Representation	Accuracy	Macro Precision	Macro Recall	Macro F1-Score
SVM (RBF)	Mean Pooling	82.99%	82.15%	76.96%	79.13%
	Norm-Based Softmax Weighting	82.42%	79.97%	80.15%	79.95%
Logistic Regression	Mean Pooling	80.91%	76.93%	80.69%	78.46%
	Norm-Based Softmax Weighting	79.96%	76.17%	80.21%	77.68%

Based on Table 5, SVM achieved higher performance than Logistic Regression for both representation strategies. On the test data, SVM with Mean Pooling achieved the highest accuracy of 82.99%, while the highest Macro F1 score was achieved by SVM with Norm-Based Softmax Weighting at 79.95%. For Logistic Regression, Mean Pooling yielded an accuracy of 80.91% and a Macro F1 score of 78.46%, which were higher than the 79.96% accuracy and 77.68% Macro F1 score achieved with Norm-Based Softmax Weighting. These results indicate that changes in representation strategies have different effects on each classifier.

Ablation Analysis

Ablation analysis was conducted to evaluate the impact of aggregation strategies on classification performance using Word2Vec with SVM and Logistic Regression. The evaluation was based on accuracy and macro F1 score on the test data. The evaluation results for both aggregation strategies are shown in Figure 5.

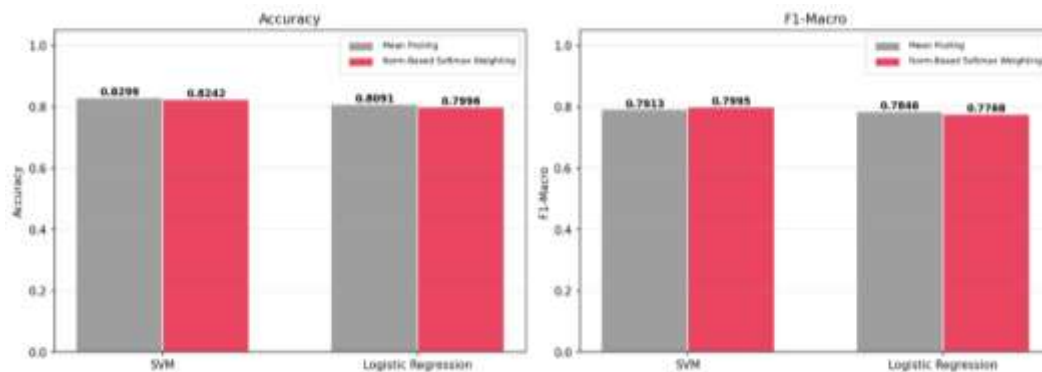


Figure 4 Ablation Comparison of Mean Pooling and Norm-Based Softmax Weighting

Based on Figure 5, in SVM, Norm-Based Softmax Weighting increased the Macro F1 score from 0.7913 to 0.7995 (+0.0082) compared to Mean Pooling, but the accuracy decreased from 0.8299 to 0.8242. In Logistic Regression, the Macro F1 score decreased from 0.7846 to 0.7768, while Accuracy decreased from 0.8091 to 0.7996. These results indicate that the impact of the aggregation strategy on performance differs between the two classifiers.

To test the significance of the performance differences, a McNemar test was conducted, and a 95% confidence interval was calculated for the difference in Macro F1. The test results are presented in Table 6.

Table 6 Ablation and Statistical Comparison of Aggregation Strategies

Model	Δ Accuracy	Δ Macro F1	95% CI Δ F1	McNemar p-value	Conclusion
SVM (RBF)	-0.0057	+0.0082	[-0.0253; 0.0416]	0.7877	Not Significant
Logistic Regression	-0.0095	-0.0078	[-0.0294; 0.0137]	0.4869	Not Significant

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

The results of the statistical tests show that the changes in performance for both classifiers are not significant. For the SVM, the increase in Macro F1 of 0.0082 has a 95% confidence interval of $[-0.0253; 0.0416]$ and a McNemar p-value of 0.7877. For Logistic Regression, the change in Macro F1 of -0.0078 has a 95% confidence interval of $[-0.0294; 0.0137]$ with $p = 0.4869$. Both confidence intervals include zero, and the p-values are greater than 0.05. Thus, based on the test data from this study, the improvement in performance of Norm-Based Softmax Weighting over Mean Pooling is not supported by statistically significant evidence.

Token Weighting Analysis

A token analysis was performed on 277 tokens with a minimum document frequency of 5 in the training data. L2 norm showed a weak positive correlation with mutual information regarding class (Spearman's $r = 0.151$; $p = 0.0119$), but correlated more strongly with document frequency ($r = 0.520$; $p < 0.001$). After controlling for document frequency, the partial correlation between L2 norm and mutual information became negative ($r = -0.200$); ($p = 0.0008$). For example, tokens such as *tunjukkan*, *apa*, and *mana* have relatively low L2 norms 3.639, 3.400, and 3.658, respectively with mutual information values of 0.112, 0.081, and 0.031. Conversely, tokens with the highest L2 norms, such as *tu*, *buru*, and *set*, have mutual information values ranging from 0.001 to 0.002. These findings suggest that L2 norm does not always correlate with a token's relevance in distinguishing intention classes.

Weighting diagnostics on 371 test comments containing at least two in-vocabulary tokens showed that the ratio of the maximum weight to the average uniform weight was 1.563. The normalized entropy reached 0.889, with a value of 1.000 indicating a uniform weight distribution. Furthermore, the cosine similarity between the vector resulting from Norm-Based Softmax Weighting and the Mean Pooling vector reached 0.9731. These results indicate that the weight distribution tends to be nearly uniform and that both strategies produce very similar representations. This finding helps explain why the performance differences between the two strategies in Table 6 are not statistically significant.

Generalization Analysis

To evaluate the model's generalization ability on videos different from the training data, an out-of-video validation was performed using the Leave-One-Video-Out (LOVO) approach. Each video was sequentially set aside as test data, while the other videos were used as training data. The evaluation used Macro F1 for each combination of classifier and representation strategy, with the results per video presented in Figure 6.

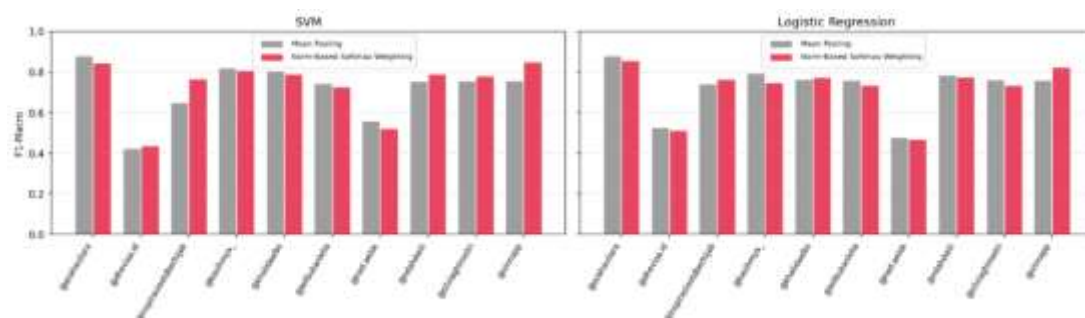


Fig. 5 Out-of-Video Validation Using Leave-One-Video-Out

Based on Figure 6, the Macro F1 score shows variations in performance across videos for both classifiers. For the SVM, the combined Macro F1 score for all LOVO predictions was 0.7579 with Mean Pooling and 0.7777 with Norm-Based Softmax Weighting, while the average Macro F1 score per video was 0.7146 ± 0.1359 and 0.7308 ± 0.1390 , respectively. For Logistic Regression, the combined Macro F1 scores were 0.7564 and 0.7569, with per-video averages of 0.7244 ± 0.1235 and 0.7184 ± 0.1265 . The Macro F1 per video ranged from 0.4231 to 0.8799 across all configurations. These results indicate that model performance may vary across the test videos, while changes in representation strategies yield distinct patterns for each classifier. For group-based validation, GroupKFold yielded a Macro F1 score of 0.7614 for SVM with Mean Pooling and 0.7755 with Norm-Based Softmax Weighting, while Logistic Regression achieved 0.7571 and 0.7555.

DISCUSSIONS

The results of the study show that the effect of Norm-Based Softmax Weighting on classification performance differs between Support Vector Machines (SVM) and Logistic Regression. In SVMs, Norm-Based Softmax Weighting improves Macro F1 compared to Mean Pooling, although Accuracy decreases slightly. Conversely, in

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Logistic Regression, the use of Norm-Based Softmax Weighting reduces both Macro F1 and Accuracy compared to Mean Pooling. The results of ablation analysis and statistical testing indicate that these differences do not yet provide sufficient evidence to conclude that one aggregation strategy is generally superior. These findings suggest that norm-based weighting produces representations that are competitive with Mean Pooling, but its contribution depends on the classifier used.

The difference in performance between SVM and Logistic Regression indicates that the effectiveness of a representation strategy is determined not only by the feature representation but also by the characteristics of the classifier used. SVM outperformed Logistic Regression under both representation strategies, both in the evaluation of the test data and in cross-validation. This finding suggests that the Word2Vec representation used in this study is more suitable when combined with SVM to form decision boundaries between intent classes. These results are consistent with previous studies showing that the use of Word2Vec with SVM is a competitive approach for classifying Indonesian-language text (Andriyani et al., 2024; Rizky & Lhaksana, 2023). Thus, the comparison of the two classifiers in this study demonstrates that changes in aggregation strategies need to be considered in conjunction with the characteristics of the classification model used.

At the class level, Opinion and Request tend to be easier to classify than Question across all model configurations. The Question class has the lowest F1-score, ranging from 0.6578 to 0.6986, while the highest F1-score is achieved for the Opinion or Request classes. In SVM, Norm-Based Softmax Weighting increased the Recall for the "Question" class from 0.6162 to 0.7374 and the F1-score from 0.6740 to 0.6986. In Logistic Regression, these changes did not follow the same pattern. These results indicate that token weighting can aid in the classification of certain classes, but its effect is not consistent across all categories. This may be related to the relatively short and informal nature of TikTok comments, meaning that the boundaries between comments expressing questions and requests may share similar linguistic characteristics.

The findings of this study provide additional insights into the use of Word2Vec and token weighting mechanisms in text classification. The use of Word2Vec in conjunction with SVMs has demonstrated competitive performance on various text classification tasks, while token weighting approaches can be used to contribute differently to text representations (Muchori & Maina, 2025; Zhao et al., 2022). This study extends these tests through ablation experiments that directly compare Mean Pooling and Norm-Based Softmax Weighting on the same classifier. However, the results of diagnostic analysis indicate that the L2 norm cannot be positioned as a direct measure of a token's semantic relevance. After controlling for frequency, the partial correlation between the L2 norm and mutual information with respect to the class is negative ($r = -0.200$), and the resulting weights approach uniform weights (cosine similarity to the Mean Pooling vector is 0.9731); thus, the mechanism used is more accurately understood as a non-learned weighting heuristic rather than a measure of semantic importance.

Video-based validation results show that the model's generalization ability is lower than that of random data. In the Leave-One-Video-Out test, the combined Macro F1 score for SVM increased from 0.7579 with Mean Pooling to 0.7777 with Norm-Based Softmax Weighting, whereas for Logistic Regression the difference was very small, at 0.7564 and 0.7569, respectively. The GroupKFold results show a similar pattern for SVM, but not for Logistic Regression. Performance variation across videos is also quite significant, with the Macro F1 score per video ranging from 0.4231 to 0.8799. Therefore, the improvement from Norm-Based Softmax in some configurations cannot yet be considered a consistent advantage. These findings suggest that the characteristics of each video can influence the model's ability to generalize to new comments. Additionally, the research dataset is limited to 10 TikTok Fashion Affiliate videos focusing on women's products; thus, the research results cannot yet be generalized to all types of TikTok content or other product categories.

CONCLUSION

This study evaluates Attention-Weighted Word2Vec with Norm-Based Softmax Weighting as a non-learned weighting mechanism for classifying the intent of TikTok Fashion Affiliate comments. The results show that Norm-Based Softmax Weighting delivers competitive performance compared to Mean Pooling, but does not consistently outperform it. In SVM, Norm-Based Softmax Weighting yields a slightly higher Macro F1 score than Mean Pooling, whereas in Logistic Regression, its performance declines. SVM also outperforms Logistic Regression for both representation strategies. Ablation results and statistical tests have not yet provided sufficient evidence to conclude that Norm-Based Softmax Weighting is generally superior.

The research findings indicate that representation strategies and classifiers need to be considered together. The "Opinion" and "Request" classes tend to be easier to classify than "Question," while diagnostic analysis shows that the L2 norm does not directly reflect the relevance of token classes. Therefore, Norm-Based Softmax Weighting is more appropriately viewed as a simple, non-learned heuristic weighting mechanism rather than a direct measure of a token's semantic importance. Video-based validation also revealed a decline in performance and inter-video variability, indicating that model generalization is still influenced by data characteristics.

This study has limitations regarding the amount and scope of the data, specifically 10 TikTok affiliate videos focused on women's fashion products. The Leave-One-Video-Out validation results show a decrease in Macro F1

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

compared to random data splitting, as well as significant performance variation across videos; thus, the model's generalization ability still needs to be examined on data with a greater variety of videos, product categories, and comment characteristics. Future research is recommended to expand the number and variety of videos as well as product categories, while also developing a weighting mechanism that more directly accounts for token-level information. Thus, this study provides an empirical evaluation of the use of non-learned weighting in Word2Vec and demonstrates that the choice of representation strategy must be considered in conjunction with the characteristics of the classifier and the data used.

REFERENCES

- Andriyani, W., Astuti, Y. and Wisesa, B.A. (2024) "Analisis Sentimen pada Ulasan Produk dengan SVM dan Word2Vec Sentiment Analysis on Product Reviews with SVM and Word2Vec," (1), pp. 173–185. Available at: <https://doi.org/10.26798/jiko.v8i1.1498>.
- Asramanggala, M.S., Prasetyowati, S.S. and Sibaroni, Y. (2023) "Optimal Number Data Trains in Hoax News Detection of Indonesian using SVM and Word2Vec," 5(1), pp. 21–28. Available at: <https://doi.org/10.47065/bits.v5i1.3516>.
- Aubaid, A.M., Mishra, Alok and Mishra, Atul (2024) "Machine learning and rule-based embedding techniques for classifying text documents," International Journal of System Assurance Engineering and Management, 15(12), pp. 5637–5652. Available at: <https://doi.org/10.1007/s13198-024-02555>.
- Ayuningtyas, P. and Tantyoko, H. (2024) "Perbandingan Metode Word2vec Model Skipgram pada Ulasan Aplikasi Linkaja menggunakan Algoritma Bidirectional LSTM dan Support Vector Machine Comparison of the Word2vec Skipgram Model Method Linkaja Application Review using Bidirectional LSTM Algorithm and Support Vector Machine," 12(1), pp. 189–196. Available at: <https://doi.org/10.26418/justin.v12i1.72530>.
- Hara, T., Kurita, H., Imaizumi, M., Inui, K., & Yokoi, S. (2026). *Why Mean Pooling Works : Quantifying Second-Order Collapse in Text Embeddings*. 1, 47180–47201. <https://doi.org/10.18653/v1/2026.acl-long.2183>
- Harahap, Y.A. and Sazali, H. (2024) "Digital Marketing Tiktok Affiliate pada Pemasaran Produk Fashion," 6(5), pp. 1733–1743. Available at: <https://doi.org/10.38035/rrj.v6i5.1003>.
- Hashmi, E. et al. (2025) "Multi-Domain Text Classification Via Linguistic And Semantic Feature Integration." Available at: <https://doi.org/10.1007/s13278-025-01551-7>.
- Hendrawan, I.R., Utami, E. and Hartanto, A.D. (2022) "Analisis Perbandingan Metode Tf-Idf dan Word2vec pada Klasifikasi Teks Sentimen Masyarakat Terhadap Produk Lokal di Indonesia," pp. 497–503. Available at: <https://doi.org/10.30591/smartcomp.v11i3.3902>.
- Kaibassova, D. (2025) "Document Analysis Via Combined Vectorization And Machine Learning Approaches," 8(4), pp. 2195–2204. Available at: <https://doi.org/10.53894/ijirss.v8i4.8356>.
- Li, J. et al. (2025) An Improved Attention-Based LSTM Neural Network for Intelligent Anomaly Detection in Financial Statements, 2025 2nd International Conference on Artificial Intelligence, Digital Media Technology and Interaction Design (ICADI 2025), November 28-30, 2025, Tianjin, China. Association for Computing Machinery. Available at: <https://doi.org/10.1145/3795926.3796015>.
- Liu, X., Lai, H., Wong, D. F., & Chao, L. S. (2020). *Norm-Based Curriculum Learning for Neural Machine Translation*. 427–436. Available at: <https://doi.org/10.18653/v1/2020.acl-main.41>.
- Maulani, R., Ula, M. and Retno, S. (2026) "Klasifikasi Teks Berbahasa Indonesia Menggunakan Word2vec Dan K-Nearest Neighbor Untuk Identifikasi Berita Hoax 1) 1,2,3)," 11(1), pp. 122–129. Available at: <https://doi.org/10.36341/rabit.v11i1.6674>
- Media, J. and Budidarma, I. (2022) "Feature Expansion Using Word2vec for Hate Speech Detection on Indonesian Twitter with Classification Using SVM and Random," 6(April), pp. 979–988. Available at: <https://doi.org/10.30865/mib.v6i2.3855>.
- Muchori, J.G. and Maina, M.P. (2025) "Review Article A Systematic Literature Review on Application of Text Embedding Techniques in NLP," 13(December), pp. 45–61. Available at: <https://doi.org/10.26438/ijsrcse.v13i6.791>.
- Priyadarshini, J. (2025) "Enhanced Hybrid Deep Learning Model With Improved Self-Attention Mechanism for Legal Judgment Prediction," IEEE Access, 13(August), pp. 139868–139882. Available at: <https://doi.org/10.1109/ACCESS.2025.3596180>.
- Pujianto, D. and Asia, U.M. (2025) "Journal of Computer Networks , Architecture and High Performance Computing Application Of Transformer Model And Word Embedding In Sentiment Analisis Of Indonesian E- Journal of Computer Networks , Architecture and High Performance Computing," 7(3), pp. 720–732.
- Riana, D. et al. (2025) "Word2vec Optimalization Using Transfer Learning in Indonesian Language for Higher Education," Jurnal Ilmu Pengetahuan dan Teknologi Komputer, 11(2), pp. 382–390. Available at: <https://doi.org/10.33480/jitk.v11i2.6051>.

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

- Rifaldy, F., Sibaroni, Y. and Prasetyowati, S.S. (2025) "Effectiveness of word2vec and tf-idf in sentiment classification on online investment platforms using support vector machine 1.," 10(2), pp. 863–874. Available at: <https://doi.org/10.29100/jipi.v10i2.6055>.
- Rinaldi, M.K. et al. (2026) "Application of Two-Stream Late Fusion on EfficientNetV2 based on Transfer Learning to classify AI-generated paintings," 10(2), pp. 976–990. Available at: <https://doi.org/10.33395/sinkron.v10i2.15814>.
- Rizky, F.M. and Lhaksana, K.M. (2023) "Twitter Sentiment Analysis of Kanjuruhan Disaster using Word2Vec and Support Vector Machine," 5(1), pp. 219–227. Available at: <https://doi.org/10.47065/bits.v5i1.3612>.
- Sahbuddin, M. and Agustian, S. (2022) "JITE (Journal of Informatics and Telecommunication Engineering) Support Vector Machine Method with Word2vec for Covid-19 Vaccine Sentiment Classification on Twitter," 6(July), pp. 288–297. Available at: <https://doi.org/10.31289/jite.v6i1.7534>.
- Saputra, J. et al. (2025) "Analisis Performa Naive Bayes dan SVM terhadap Sentimen Teks Media Sosial dengan Word2Vec dan SMOTE," 10(1), pp. 143–155. Available at: <https://doi.org/10.24252/instek.v10i1.54889>.
- Science, C. and Fauzi, M.A. (2019) "Word2Vec model for sentiment analysis of product reviews in Indonesian language," 9(1), pp. 525–530. Available at: <https://doi.org/10.11591/ijece.v9i1.pp525-530>.
- Suharman, A. et al. (2025) "Analisis Sentimen Pengguna Aplikasi Livin' by Mandiri Menggunakan Metode Support Vector Machine (SVM) dengan Ekstraksi Fitur TF-IDF dan Word2Vec User Sentiment Analysis of the Livin' by Mandiri Application Using the Support Vector Machine (SVM) Method with TF-IDF and Word2Vec Feature Extraction," 5(8), pp. 2201–2212. Available at: <https://doi.org/10.52436/1.jpti.941>.
- Wicaksana, H.S. and Huda, K. (2025) "Penerapan Word2Vec dan SVM dengan Hyperparameter Tuning untuk Deteksi Phishing," 12(3), pp. 361–371. Available at: <https://doi.org/10.30865/jurikom.v12i3.8729>.
- Yousefi, R. et al. (2026) "Enhancing Logistic Regression Classification : Insights From Simulation And Real-World Applications Through Ranked Set Sampling," pp. 1–28. Available at: <https://doi.org/10.1038/s41598-026-41333-5>.
- Zhao, W. et al. (2022) "WTL-CNN : a News Text Classification Method Of Convolutional Neural Network Based On Weighted Word Embedding," 0091. Available at: <https://doi.org/10.1080/09540091.2022.2117274>.