Published online on the journal's webpage: **http://journals.telkomuniversity.ac.id/index.php/dinda**

Journal of Dinda

Data Science, Information Technology, and Data Analytics

Vol. 6 No. 2 (2026) 71 - 78 E-ISSN: 2809-8064

Emotion Classification of YouTube Comments Using Cost Sensitive Support Vector Machine

Desi Daomara Sitanggang[1*], Wahyu Syaifullah Jauharis Saputra[2*], Muhammad Nasrudin[3]

[1,2,3]Sains Data, Fakultas Ilmu Komputer, UPN "Veteran" Jawa Timur, Indonesia

[1*]desidaomara@gmail.com, [2]wahyu.s.j.saputra.if@upnjatim.ac.id, [3]nasrudin.fasilkom@upnjatim.ac.id

Abstract

YouTube is the most widely used digital platform in the world. Its comment section serves as a rich space for emotional expression on music content. The song Takut by Idgitaf has gained over 58 million views. It generated thousands of public responses with diverse emotional reactions. Previous studies focused on song lyrics rather than public responses. Conventional methods such as TF-IDF fail to capture the semantic context of informal language. This study aims to classify emotions in YouTube comments into eight categories based on the NRC Emotion Lexicon. A total of 13.462 comments were collected via the YouTube API v3. Preprocessing included cleaning, case folding, normalization, tokenization, stopword removal, and stemming. Labeling used the NRC Emotion Lexicon automatically. This process resulted in 7.239 labeled data points. Manual validation of 100 random samples showed a 35 percent agreement rate. Feature extraction employed Word2Vec with a 200 dimension Skip gram architecture to capture semantic context. TF-IDF served as a baseline comparison. Classification used Cost-Sensitive Support Vector Machine to handle class imbalance. The Word2Vec model with Linear kernel achieved 70.72 percent accuracy. The TF-IDF baseline achieved 80.52 percent accuracy. TF-IDF outperformed Word2Vec due to its exact match alignment with the rigid lexicon labeling. Cost penalty parameters proved effective in maintaining model sensitivity toward minority emotion classes.

Keywords: *Cost-Sensitive Support Vector Machine, Emotion Classification, NRC Emotion Lexicon, Word2Vec, YouTube.*

1. Introduction

The advancement of technology has brought significant changes in human life, particularly in the way people communicate and express their opinions through digital platforms. We Are Social data in 2025 shows that YouTube is the most widely used platform in the world [1]. YouTube not only functions as an entertainment medium, but also as a space for public emotional expression that allows users to interact freely through the comment section [2]. In the context of musical works, song lyrics with profound messages are capable of creating high emotional engagement, encouraging listeners to openly express their feelings in the digital space [3]. This phenomenon is evident in the music video of the song "Takut" by Idgitaf, a music video has reached over 58 million views and attracted thousands of public responses containing diverse and complex emotional nuances, forming a rich textual dataset for exploration. However, the large volume of data makes manual analysis highly inefficient and impractical.

Several studies have conducted emotion classification on songs, but the analysis has been limited to song lyrics only [4][5]. Research on emotion classification that examines public responses to a song remains very limited. Emotion classification is the process of identifying and determining the type of emotion contained in a sentence or paragraph, as well as predicting emotions from text data accurately, enabling deeper understanding of sentiment and better decision-making based on emotion expression [6]. Many researchers use the Support Vector Machine (SVM) algorithm as it has proven effective in classifying high-dimensional text data. Research [7] applied several algorithms comparing SVM, Naïve Bayes, and LSTM on YouTube comments, proving that SVM outperforms the other two. The choice of kernel type in SVM also significantly affects classification accuracy [8]. However, these studies generally still use traditional feature extraction such as TF-IDF, which ignores the semantic context of words. Research [9] integrated SVM with Word2Vec to capture semantic similarity between informal words in text. Another common problem in text

Received: 28-06-2026 | Revised: 22-07-2026 | Published: 21-08-2026

analysis is class imbalance. Research [10] attempted to address this using SMOTE, but this oversampling technique risks distorting the original context of comment sentences. As an alternative solution, [11] applied a Cost-Sensitive Support Vector Machine (CS-SVM) approach, which proved effective in improving minority class detection without manipulating the original data.

Previous studies on music often focus on the sentiment analysis of song lyrics. Analyzing lyrics alone does not capture the actual emotional impact on listeners. Several studies have explored listener responses through social media. These studies predominantly use basic sentiment analysis to determine positive or negative polarity. These previous studies also heavily rely on traditional statistical feature extraction. This study fills the research gap by focusing on the emotion classification of YouTube comments. These comments represent direct listener responses. This research categorizes eight specific emotions using the NRC Emotion Lexicon. This study also implements Word2Vec word embedding. This modern method captures complex semantic relationships in Indonesian text better than traditional methods. To empirically validate this, this study compares Word2Vec against TF-IDF as a baseline feature extraction method.

Therefore, this study aims to classify emotions in YouTube comments on the song Takut by Idgitaf into eight emotion categories based on the NRC Emotion Lexicon. This research offers novelty by combining the Cost Sensitive Support Vector Machine algorithm with Word2Vec feature extraction to classify informal Indonesian text. This combination directly addresses the extreme data imbalance and captures semantic relationships better than traditional TF-IDF methods. The study also compares the performance of Linear, Polynomial, RBF, and Sigmoid kernels to find the most optimal model. This research contributes to the development of text classification models and provides practical insights for the music industry regarding listener sentiment.

2. Research Methods

This study uses a quantitative approach with a comparative experimental method. The method applied in this study is Cost-Sensitive Support Vector Machine (CS-SVM). The overall research stages and workflow are show in Figure 1.

2.1. Data Collection

The researchers collected secondary data from the YouTube platform by crawling data using the YouTube API v3. The data was extracted from the music video for the song “Takut” with the video ID “5TUUg9mU_V0”.

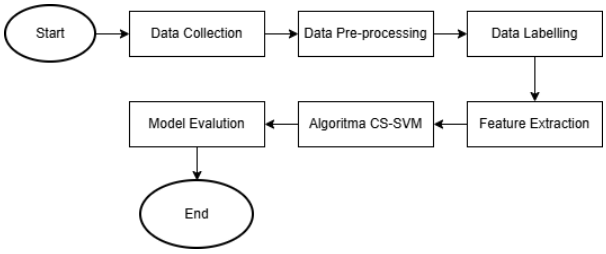

Figure 1. Flowchart

The data collection period ran from the release date of the video until May 29 2026. This process successfully collected 13462 raw comments. The researchers applied a language filter to ensure all data used Indonesian. The researchers also removed duplicate comments and excluded comment replies to maintain the originality of individual responses. The researchers anonymized user names across all data to comply with ethical considerations for public data privacy.

2.2. Data Preprocessing

The preprocessing stage was conducted to clean the data from various disturbances so that the text data is ready for optimal analysis [12]. This process includes several steps. Cleaning removes usernames, URLs, emojis, symbols, numbers, and other unnecessary elements. Case folding converts all uppercase letters to lowercase. Normalization replaces non-standard slang and mixed Indonesian-English words with standard forms and reduces repeated characters. Tokenization splits text into individual units. Stopword removal deletes words that carry no significant meaning. Stemming reduces words to their root forms. These steps improve data quality and consistency for an effective classification process. The specific examples of text changes for each step are presented in the Table 1.

2.3. Data Labeling

The data labeling stage was performed automatically using the Indonesian translated version of the NRC Emotion Lexicon. This dictionary was chosen because it has a broad vocabulary coverage. This coverage is suitable for capturing various emotional expressions in Indonesian text. The lexicon contains 14182 words mapped into 8 emotion columns and 2 sentiment columns [13]. The system matches each word in the cleaned comments with the lexicon vocabulary. This step determines the dominant emotion. The system evaluates the emotional weight of each word during this process. The algorithm detected 5050 comments with a total score of zero across all eight emotion categories.

Table 1. Preprocessing

Comment	Cleaning	Case Folding	Normalization	Tokenization	Stopword	Stemming
Terharu banget sama semua komentar disiniii, b...	Terharu banget sama semua komentar disiniii bi...	terharu banget sama semua komentar disiniii bi...	terharu banget sama semua komentar disini biki...	['terharu', 'banget', 'sama', 'semua', 'koment...	['terharu', 'banget', 'komentar', 'bikin', 'be...	haru banget komentar bikin berat juang thankyo...
Terharu, senang, baca cerita-cerita teman-tema..	Terharu senang baca ceritacerita temanteman Se...	terharu senang baca ceritacerita temanteman se...	terharu senang baca ceritacerita temanteman se...	['terharu', 'senang', 'baca', 'ceritacerita', ...	['terharu', 'senang', 'baca', 'ceritacerita', ...	haru senang baca ceritacerita temanteman semangat
Semangat!!	Semangat	semangat	semangat	['semangat']	['semangat']	semangat
Lagunya bagus banget❤️	Lagunya bagus banget	lagunya bagus banget	lagunya bagus banget	[lagunya, bagus, banget]	[lagunya, bagus, banget]	lagunya bagus banget❤️

For example, a comment like "mantul bgtlgu nih" receives a score of zero across all emotion columns and is categorized as neutral. The system classified these comments as neutral or lacking emotional content. The system then removed these neutral comments from the dataset. This process left 7239 valid data points. The system classified these valid data into anger, anticipation, disgust, fear, joy, sadness, surprise, and trust.

To validate the automatic labeling process, a random sample of 100 comments was manually reviewed by human. The manual validation resulted in an agreement rate of only 35 percent with the NRC Emotion Lexicon. This low agreement rate indicates that word level lexicon matching struggles to accurately capture the true emotional context of informal Indonesian comments. Informal comments frequently contain slang, implicit meanings, and complex semantics that a rigid dictionary cannot translate effectively.

2.4. Feature Extraction

After the labeling stage, the text data was converted into numerical vector representations using Word2Vec. This technique is called word embedding. This method is used because words with similar meanings will be mapped to nearby points in vector space. It captures the syntactic and semantic meaning of words using NLP, unlike TF-IDF which only calculates statistical word frequency [14]. Word2Vec has two types of architecture, namely Continuous Bag of Words (CBOW) and Skip-gram. This study uses the Skip-gram architecture. The Word2Vec model was trained from scratch using the comment dataset to capture specific textual contexts. The model parameters were configured with a vector size of 200 to represent word dimensions. The window size was set to 5 to capture surrounding word contexts. The minimum word count was set to 3 to filter out rare words. The model was trained for 50 epochs using the default learning rate of 0.025. The final document embeddings were generated by calculating the average of the word vectors within each comment. Mathematically, the objective function is defined as follows:

$$Maximize\ \frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c, j\neq 0}\log P\left(w_{t+j}|w_t\right) \quad (1)$$

In its calculation, the conditional probability $p(w_{t+j} + w_t)$ or the likelihood of the context word w_O appearing given the know target word (w_I), is calculated using the following softmax function:

$$p(w_O|w_I) = \frac{\exp\left({v'_{w_O}}^T v_{w_I}\right)}{\sum_{w=1}^{V}\exp\left({v'_{w_O}}^T v_{w_I}\right)} \quad (2)$$

The final result of the Word2Vec calculation produces a weight matrix vector for each word. To measure the degree of similarity between two words from the weight vectors, cosine similarity is commonly used by measuring the cosine angle between two vectors:

$$Cosine\ Similarity = \cos(\theta) = \frac{A\,.\,B}{\|A\|\|B\|} \quad (3)$$

A cosine similarity value close to 1 indicates that the two words have very similar meanings and usage contexts in the corpus.

Word2Vec produces vector representations at the word level. The SVM classifier requires a single vector representation for each comment document. This research uses the Average Pooling method to generate document vectors. The system calculates the average value of all word vectors contained in one comment. If a comment does not contain words recognized by the Word2Vec model, the system assigns a zero vector. This aggregation method ensures each comment has a uniform vector dimension before entering the classification stage. To evaluate the effectiveness of

Word2Vec, this study also implemented TF-IDF as a baseline. TF-IDF calculates the statistical frequency of words to allow an empirical comparison against the semantic based approach of Word2Vec.

2.5. CS-SVM Algorithm

The classification model in this study was built using the Cost-Sensitive Support Vector Machine (CS-SVM) algorithm. This algorithm modifies the standard Support Vector Machine model to handle class imbalance across the eight emotion categories [15]. The system works by assigning different penalty values to each class when misclassification occurs [16]. The optimization function od CS-SVM is mathematically defined as:

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C^+ \sum_{i \in y^+} \xi_i + C^- \sum_{i \in y^-} \xi_i \quad (4)$$

Subject to:

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1,2, \dots, n \quad (5)$$

Where:

$\frac{1}{2}\|w\|^2$: the margin to be maximized
ξ_i : Slack variable for data point *i* on the wrong side of the margin
C^+ : Cost penalty parameter for misclassification of minority class
C^- : Cost penalty parameter for misclassification of majority class
x_i : Word2Vec vector representation of comment *i*
y_i : actual emotion label of comment *i*

The CS-SVM classification model used specific configurations to handle imbalanced data. The model applied the balanced class weight parameter. This mechanism calculated class weights inversely proportional to class frequencies in the input data. Minority classes automatically received higher penalty weights. The model also set the cost penalty parameter C to a constant value of 100. This configuration forced the model to strictly penalize misclassifications across all emotion categories. These parameter configurations were applied consistently across all kernel types including linear, polynomial, RBF, and sigmoid.

2.6. Model Evaluation

The researcher divided the dataset into training and testing sets to evaluate model performance. The evaluation relied on a direct split method. The data split utilized an 80 to 20 ratio. The system allocated 80 percent of the data for training and 20 percent for testing. The system applied a random seed of 42 to ensure the reproducibility of the data split. The system also used stratified sampling based on the emotion labels. This stratified method maintains the original proportion of the eight emotion categories in both the training and testing sets.

Model evaluation is the process of evaluating model performance using a confusion matrix. The evaluation metrics measure the success rate of the model based on accuracy, precision, recall, and F1-score for each kernel type. Accuracy measures how precisely the model performs classification. Precision calculates the ratio of true positives compared to the total predicted positives. Recall calculates the true positive predictions compared to the total actual positives. F1-score calculates the balance between precision and recall [17]. The accuracy and confusion matrix can be formed as in equations (6), (7), (8), (9).

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad (6)$$

$$Presisi = \frac{TP}{TP + FP} \quad (7)$$

$$Recall = \frac{TP}{TP + FN} \quad (8)$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (9)$$

3. Results and Discussion

The first step in this study was collecting data from YouTube comments. The data collection process obtained 13,462 raw comments. Table 2 shows a sample of this raw data. This dataset requires cleaning and labeling before the modeling stage.

Table 2. Dataset Sample

	Comment
0	Terharu banget sama semua komentar disiniii, bikin aku ngerasa ternyata aku ga sendirian, ternyata masih banyak yang masalahnya lebih berat, ternyata aku berjuang ga sendiri, Thankyou Kak Gita, lagunya indah sekali! Thankyou sharingnya temen2 semua, kita semua hebat, kita semua pasti bisa capai apa yang sedang diperjuangkan!:&#39
1	Terharu, senang, baca cerita-cerita teman-teman. Semangat buat semuanya. Kita gak sendirian! :&#39
2	Semangat!!
3	🥺❤️

The preprocessing stage applied cleaning, case folding, normalization, tokenization, stopword removal, and stemming to the raw text. This stage successfully produced 12,289 clean comments. The system then executed automatic labeling with the NRC Emotion

Lexicon to map words into emotion categories. The mapping results showed 5,050 comments had a zero weight across all categories. The system identified these texts as neutral comments and removed them from the dataset. This process produced 7,239 valid data points with emotional content. The system classified these data points into eight basic emotions. Table 3 presents the data distribution for each emotion category.

Table 3. Distribution Label

Label Emotion	Distribution
Trust	1.758
Fear	1.542
Anticipation	1.519
Joy	836
Sadness	631
Anger	575
Disgust	291
Surprise	87

Table 3 shows a significant data imbalance across classes. The trust, fear, and anticipation emotion classes dominate the dataset. The surprise emotion class is the smallest minority class. This data imbalance is the main reason for applying the Cost-Sensitive Support Vector Machine algorithm. This algorithm handles the dominance of the majority class. The model remains capable of detecting minority classes optimally. The next stage is feature extraction using Word2Vec. The system converts labeled text into numerical vector representations. After labeling stage, the next step is the feature extraction stage using Word2Vec. In this stage, labeled text data is converted into numerical vector representations. Table 3 displays a sample of the Word2Vec results. The system represents each comment in the form of a 200-dimensional numerical vector. This numerical matrix becomes the input data in the training process of the Cost-Sensitive Support Vector Machine classification model.

Table 4. Word2Vec Result

	Steming_data	Emotion_label	Word2Vec
0	haru banget komentar bikin berat juang thankyo...	Joy	[0.53192276, 0.17925057, 0.6034695, 0.60234165...]
1	haru senang baca ceritacerita temanteman semangat	Joy	[0.7231241, 0.3125841, 0.1969769, 0.591706, 0....]
2	semangat	Anticipation	[1.2613093, 0.44580838, 0.9100908, 0.74530256,...]
3	relevan banget sih lagu bimbang tentu kuiahker...	Fear	[-0.08472719, 0.16609621, 0.62452793, 0.245015...]

The system evaluates the performance of the Cost-Sensitive Support Vector Machine algorithm using four kernel functions. Table 4 displays a comparison of evaluation metrics for the Linear, Polynomial, Radial Basis Function, and Sigmoid kernels. Measurement uses weighted average calculation for precision, recall, and F1-score values.

Table 5. Model Evalution

Kernel	Accuracy	Precision	Recall	F1-Score	Macro Avg F1-Score
Linear	70,72%	72%	71%	71%	63%
Polynomial	64,99%	65%	65%	65%	58%
RBF	68,78%	69%	69%	69%	63%
Sigmoid	31,77%	44%	32%	34%	31%

The test results show that the Linear kernel produces the best performance. This kernel achieves the highest accuracy rate of 70.72 percent. The model evaluation also considers the macro average F1-score to ensure fairness across imbalanced classes. The Linear and RBF kernels achieved the highest macro average score of 63 percent. The Linear kernel remains the most optimal model overall due to its superior accuracy. The Sigmoid kernel recorded the worst performance across all metrics.

The Sigmoid kernel recorded the lowest performance with an accuracy of 31.77 percent. This low performance occurs due to the interaction between the Sigmoid mapping function and Word2Vec vectors. Word2Vec produces dense and continuous numerical features. The Sigmoid kernel is highly sensitive to hyperparameter tuning. The penalty parameter C of 100 used in this study does not fit the Sigmoid activation function. The Sigmoid kernel fails to map the complex semantic space of YouTube comments into a separable hyperplane. This proves that the Sigmoid kernel is less suitable for high dimensional text classification compared to the Linear kernel.

To empirically compare the feature extraction methods, the CS-SVM Linear kernel was also trained using TF-IDF. The TF-IDF model achieved a higher accuracy of 80.52 percent compared to Word2Vec at 70.72 percent. This performance gap is caused by the data labeling method. The NRC Emotion Lexicon assigns labels based on exact keyword matching. TF-IDF extracts features by counting these exact word frequencies, creating a perfect alignment. In contrast, Word2Vec converts words into semantic context vectors, which obscures the specific

keywords relied upon by the lexicon. This limitation is supported by the manual validation result, which showed only a 35 percent agreement rate between human annotators and the lexicon. The lexicon provides rigid and sometimes inaccurate labels for informal text, making it highly compatible with statistical TF-IDF but incompatible with semantic Word2Vec. Despite this, Word2Vec is maintained to capture the semantic context of informal language. This proves that lexicon based automatic labeling is more compatible with statistical extraction like TF-IDF.

Table 6. Model Comparison TF-IDF and Word2Vec

Feature Extraction	Accuracy
Word2Vec	70,72%
TF-IDF	80,52%

The application of the cost penalty in this method successfully handles the data imbalance problem. Table 7 presents the detailed classification report for the Linear kernel to provide a transparent view of the model performance across all eight emotion categories.

Table 7. Classification Report per class

Emotion	Precision	Recall	F1-Score
Anger	53%	70%	60%
Anticipation	73%	74%	73%
Disgust	49%	66%	56%
Fear	82%	77%	80%
Joy	62%	67%	64%
Sadness	67%	69%	68%
Surprise	25%	35%	29%
Trust	83%	66%	74%

Based on Table 7, the class penalty system functions to maintain model sensitivity toward minority emotions. The model still predicts the minority class surprise with a recall rate of 35 percent. The model achieved the highest precision on the trust class at 83 percent and the highest F1-score on the fear class at 80 percent. These results prove the effectiveness of the cost-sensitive approach across all emotion categories.

Based on Figure 2, the confusion matrix of the Cost-Sensitive Support Vector Machine model using the Linear kernel is displayed. This model achieves the highest accuracy rate of 70,72%. The model records the highest correct predictions on the “fear” emotion class. The system successfully predicted 239 data points correctly in that class. The model also remains capable of recognizing the minority class “surprise” with 6 correct predictions.

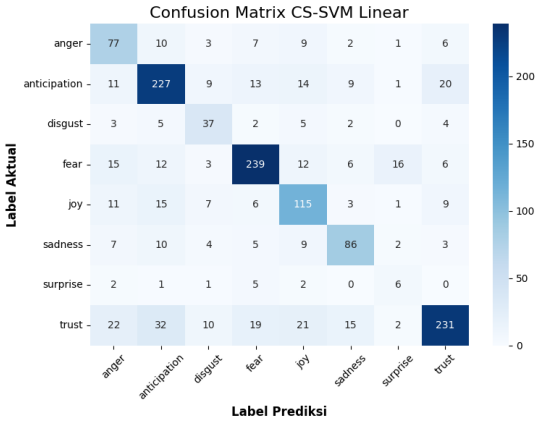

Figure 2. Confusion Matrix

To validate the effectiveness of the cost-sensitive approach, this study compares the CS-SVM model with three baseline models: Standard SVM, Random Forest, and Naive Bayes. Table 8 presents the performance comparison. While the Standard SVM achieved a similar overall accuracy of 71 percent, it performed poorly on the minority class. The Standard SVM only achieved an 18 percent recall for the surprise emotion. In contrast, the CS-SVM model maintained a competitive accuracy of 70.72 percent while significantly improving the recall of the minority surprise class to 35 percent. Random Forest and Naive Bayes models demonstrated inferior performance across all metrics. These results prove that the CS-SVM approach is effective and necessary for handling extreme class imbalance in YouTube comments.

Table 8. Model Comparison

Model	Accuracy	Macro Avg F1-score	Recall (Class Surprise)
CS-SVM	70,72%	63%	35%
SVM	71%	63%	18%
Random Forest	61%	51%	12%
Naïve Bayes	39%	35%	35%

The Linear kernel outperforms others because Word2Vec embeddings naturally form a high-dimensional space where emotion classes are often linearly separable, whereas complex kernels like Polynomial tend to overfit. Classification difficulty varies due to vocabulary distinctiveness; emotions like fear and trust are more accurately predicted because they possess distinct keywords and larger sample sizes, while surprise is the most difficult to classify due to overlapping semantic contexts with other emotions. Furthermore, extreme class imbalance, which naturally favors majority classes, is effectively mitigated by the cost-sensitive penalty. By assigning higher weights to minority categories, the model sacrifices some overall precision to maintain a 35 percent recall for the surprise

class, demonstrating the necessity of the CS-SVM approach for imbalanced YouTube comment data.

4. Conclusion

The application of the Cost Sensitive Support Vector Machine successfully classified emotions in YouTube comments. The NRC Emotion Lexicon automatically labeled eight emotion classes. Manual validation revealed a 35 percent agreement rate. This rate shows the limitation of rigid dictionary matching on informal text. The 200dimension Word2Vec feature extraction captured the semantic context of the comments. The Word2Vec model using the Linear kernel produced an accuracy rate of 70.72 percent. The TF-IDF baseline achieved 80.52 percent. The higher accuracy of TF-IDF demonstrates its strong compatibility with the exact keyword matching of the NRC Emotion Lexicon. Word2Vec focuses on broader semantic contexts. The system predicted the fear emotion class most accurately with 239 correct data points. The cost penalty successfully handled the data imbalance problem. The model detected the minority class surprise with 6 correct predictions. Future research using Word2Vec should employ manual labeling to maximize the semantic context extraction.

References

[1] S. Kemp, "Digital 2025 April Global Statshot Report." [Online]. Available: https://wearesocial.com/id/blog/2025/04/digital-2025-april-global-statshot-report/

[2] K. B. Sirait, V. Riffan, and C. G. Wijaya, "Analisis Sentimen Komentar Toxic pada Video Musik YouTube Menggunakan Metode Naive Bayes," *J. Minfo Polgan(JMP)*, vol. 15, 2026.

[3] Q. L. A. Lubis and A. A. Huda, "Implementation of SVM Algorithm to Predict Song Popularity based on Sentiment Analysis of Lyrics," *J. Appl. Informatics Comput.*, vol. 9, no. 2, pp. 265–272, 2025, doi: 10.30871/jaic.v9i2.8978.

[4] K. Z. Abidin, A. Setyanto, and R. Arief, "Systematic Literature Review terhadap Klasifikasi Emosi pada Lirik Lagu Berbahasa Ambon menggunakan Metode Bidirectional LSTM dengan Glove Word Representation Weighting," *J. Ilm. Komput. Graf.*, vol. 16, no. 2, pp. 235–244, 2023.

[5] L. W. Astuti, A. R. C, and Y. Lukito, "IMPLEMENTASI ALGORITMA NAÏVE BAYES MENGGUNAKAN ISEAR UNTUK KLASIFIKASI EMOSI LIRIK LAGU BERBAHASA INGGRIS," *J. Inform.*, vol. 14, no. 1, pp. 16–21, 2017, doi: 10.9744/informatika.14.1.16-21.

[6] M. Roberta, "KLASIFIKASI EMOSI PADA DATA TEKS PIDATO POLITIK MENGGUNAKAN," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 13, no. 1, 2026.

[7] E. L. Utari, S. H. Wibowo, T. Informatika, U. M. Bengkulu, N. Bayes, and L. Short-, "ANALISIS KOMPARATIFALGORITMASVMNAIVE BAYES DAN LSTM PADA SENTIMEN KOMENTAR LAGU LABOUR," pp. 1276–1286, 2025.

[8] T. M. Permata Aulia, N. Arifin, and R. Mayasari, "Perbandingan Kernel Support Vector Machine (Svm) Dalam Penerapan Analisis Sentimen Vaksinisasi Covid-19," *SINTECH (Science Inf. Technol. J.*, vol. 4, no. 2, pp. 139–145, 2021, doi: 10.31598/sintechjournal.v4i2.762.

[9] W. Andriyani, Y. Astuti, and B. A. Wisesa, "Analisis Sentimen pada Ulasan Produk dengan SVM dan Word2Vec Sentiment Analysis on Product Reviews with SVM and Word2Vec," *JIKO (JURNAL Inform. DAN KOMPUTER)*, no. 1, pp. 173–185, 2024, doi: 10.26798/jiko.v8i1.1498.

[10] S. Rabbani, D. Safitri, and N. Rahmadhani, "Comparative Evaluation of SVM Kernels for Sentiment Classification in Fuel Price Increase Analysis Perbandingan Evaluasi Kernel SVM untuk Klasifikasi Sentimen dalam Analisis Kenaikan Harga BBM," vol. 3, no. October, pp. 153–160, 2023.

[11] R. Zeng, Y. Lu, S. Long, C. Wang, and J. Bai, "Cardiotocography signal abnormality classification using time-frequency features and Ensemble Cost-sensitive SVM classifier," *Comput. Biol. Med.*, vol. 130, no. January, p. 104218, 2021, doi: 10.1016/j.compbiomed.2021.104218.

[12] F. O. Vianda and F. P. Putra, "Sistem Klasifikasi Komentar Media Sosial untuk Deteksi Dini Cyberbullying Menggunakan TF-IDF dan Support Vector Machine," *J. INSTEK*, vol. 11, no. 1, pp. 133–142, 2026.

[13] S. Wulandari, D. Novita, A. Wilson, P. Studi, and T. Informatika, "Analisis Sentimen dan Deteksi Emosi Menggunakan Metode NRC Emotion Lexicon Pada Google Maps Review di Wulan Rent Car," *Pros. Semin. Nas. Sains*, vol. 6, no. 1, pp. 335–344, 2025.

[14] M. D. Rhman, A. Djunaidy, and F. Mahananto, "Penerapan Weighted Word Embedding pada Pengklasifikasian Teks Berbasis Recurrent Neural Network untuk Layanan Pengaduan Perusahaan Transportasi," vol. 10, no. 1, 2021.

[15] D. L. FREIRE, "Cost-Sensitive Algorithms for Text Classification in the Legal Domain : Addressing Imbalanced Lawsuit Themes," *Trends Comput. Appl. Math.*, vol. 26, 2025, doi: 10.5540/tcam.2025.026.e01859.

[16] R. Guido, M. Carmela, and G. Domenico, "A hyper-parameter tuning approach for cost-sensitive support vector machine classifiers," *Soft Comput.*, vol. 27, no. 18, pp. 12863–12881, 2023, doi: 10.1007/s00500-022-06768-8.

[17] I. Amal, "Perbandingan Pelabelan Otomatis Dan Manual Untuk Analisis Sentimen Terhadap Kenaikan Harga BBM Pertamina Pada Twitter Menggunakan Algoritma Support Vector Machine," pp. 473–487, 2023.