Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14513
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Enhancing Sentiment Analysis Accuracy Using SVM and Slang Word Normalization on YouTube Comments

Alfin Nur Aziz Saputra[1)*], Rujianto Eko Saputro[2)], Dhanar Intan Surya Saputra[3)]
[1,2)] Department of Computer Science Universitas Amikom Purwokerto, Indonesia,
[3)]Department of Informatics Universitas Amikom Purwokerto, Indonesia
[1)]lordsaputra@gmail.com, [2)]rujianto@amikompurwokerto.ac.id, [3)]dhanar.amikom@gmail.com

Submitted :March 1, 2025 | **Accepted** : March 31, 2025 | **Published** : April 15, 2025

Abstract: This study explores enhancing sentiment analysis accuracy on YouTube comments by integrating Support Vector Machine (SVM) with slang word normalization. The dataset consists of 3,375 comments on the movie *Pengabdi Setan 2: Communion*, containing informal language, slang, and emojis. Pre-processing techniques, including text cleaning, tokenization, stopword removal, and stemming, were employed to refine the dataset for analysis. Slang words were normalized using a dictionary to replace non-standard terms with their formal counterparts. Feature extraction was performed using TF-IDF, and sentiment analysis was conducted using the VADER method, with further improvements made through SMOTEENN to balance the class distribution. After hyperparameter tuning using Grid Search, the SVM model achieved optimal performance, with precision, recall, and F1-scores of 100% for both positive and negative classes. The results indicate that combining text normalization and machine learning models can effectively handle the complexities of informal language in user-generated content, leading to highly accurate sentiment classification.

Keywords: Sentiment Analysis, Slang Word Normalization, Support Vector Machine, Text Processing, YouTube Comments

INTRODUCTION

Sentiment analysis plays a crucial role in understanding public opinion, particularly on social media platforms like YouTube. By employing natural language processing techniques, sentiment analysis enables the extraction of attitudes and emotions from user-generated content, which is abundant on these platforms (Kularbphettong et al., 2024). This computational approach allows researchers and organizations to gauge public sentiment regarding various topics, including environmental sustainability and public policies, thus facilitating informed decision-making (Lestari & Anugrahni, 2021). For instance, sentiment analysis has been effectively utilized to analyze public reactions to government initiatives and brand perceptions, providing insights that can enhance service delivery and policy formulation (Omar & Hamouda, 2021). As social media continues to grow, the ability to analyze vast amounts of user-generated content becomes increasingly valuable for understanding societal trends and public sentiment (Redjeki & Widyarto, 2022).

Analyzing informal, slang, and emotional language in user comments presents significant challenges for sentiment analysis. The informal nature of social media language, characterized by the use of slang, emoticons, and unconventional punctuation, complicates the interpretation of sentiment. For instance, VADER (Valence Aware Dictionary for Sentiment Reasoning) is specifically designed to handle such informalities, allowing for context-aware sentiment analysis by interpreting negation and disambiguating meanings based on textual clues (Rao et al., 2020). However, the diversity of language expressions and the rapid evolution of slang can still hinder accurate sentiment classification (Xiong et al., 2024). Moreover, the emotional intensity conveyed in comments can vary widely, making it difficult for traditional sentiment analysis tools to capture nuanced sentiments effectively. Studies have shown that sentiment analysis methods must adapt to the dynamic nature of user-generated content, incorporating advanced natural language processing techniques to improve accuracy (Xiong et al., 2024). Thus, while sentiment analysis is a powerful tool for understanding public opinion, it must continually evolve to address the complexities of informal language and emotional expression in user comments.

*name of corresponding author

e-ISSN : 2541-2019
p-ISSN : 2541-044X

NLP technology significantly enhances the accuracy of sentiment analysis by enabling machines to understand and interpret the nuances of human language. NLP techniques facilitate the extraction of sentiment from text by analyzing emotional tones, context, and linguistic structures, which is essential for accurate sentiment classification (Zhan et al., 2024). For instance, advanced models like GPT-3 leverage large language models (LLMs) to improve sentiment detection by understanding context and subtleties in user comments, thereby increasing the reliability of sentiment analysis outcomes (Zhan et al., 2024). Moreover, the application of machine learning algorithms, such as Support Vector Machines (SVM) and Random Forest, in conjunction with NLP techniques allows for the classification of sentiments into distinct categories, such as positive and negative (Syafia et al., 2023). This combination not only enhances the precision of sentiment predictions but also enables the analysis of emotional patterns within user-generated content, such as YouTube comments (Syafia et al., 2023). Furthermore, NLP's ability to process vast amounts of data in real-time makes it invaluable for sentiment analysis in dynamic environments like social media, where language evolves rapidly (Prestianta, 2021). Thus, NLP serves as a cornerstone for improving the accuracy and effectiveness of sentiment analysis across various platforms.

The limitations of classification methods in capturing emotional meaning and sarcasm in informal language are significant due to the inherent complexities of human communication. Sarcasm often employs positive language to convey negative sentiments, complicating sentiment analysis and requiring advanced models for accurate detection (Khan et al., 2023; Kumar et al., 2020). Traditional classification approaches struggle to interpret this intentional ambiguity, as they typically rely on explicit contextual cues, which are often absent in informal text (Bagate & Suguna, 2021). Furthermore, the reliance on binary classification models can oversimplify the nuanced emotional expressions present in informal language, leading to misinterpretations (Potamias et al., 2020). Recent advancements, such as transformer-based models, have shown promise in addressing these challenges, yet they still face difficulties in capturing the full spectrum of emotional meaning (Williams et al., 2024). Thus, while progress has been made, the task of effectively classifying sarcasm and emotional nuances in informal language remains a complex and ongoing challenge.

To enhance the accuracy of sentiment analysis models, it is crucial to adopt a multifaceted approach that incorporates advanced methodologies and rigorous preprocessing techniques. Recent studies emphasize the importance of using transformer-based models, such as BERT, which have demonstrated state-of-the-art performance in various natural language processing tasks, including sentiment analysis (Chinedu et al., 2023). Furthermore, the choice of features plays a significant role in the predictive accuracy of sentiment classification. Techniques such as Bag of Words, TF-IDF, and N-grams are essential for effective feature representation (Mutinda et al., 2021). Additionally, the preprocessing phase, which includes data crawling, labeling, and sentiment evaluation, is vital for achieving optimal results (Munggaran et al., 2023). The context of the data, particularly in politically charged environments, can also influence sentiment interpretation, necessitating tailored preprocessing methods to enhance precision (Nurodin & Puspitarani, 2023). Overall, integrating these strategies can significantly improve the representativeness and accuracy of sentiment analysis models (Chamekh et al., 2022).

This research aims to develop a more accurate sentiment analysis model by combining Support Vector Machine (SVM) and SMOTEENN data balancing techniques, which aims to address class imbalance in sentiment data. In addition, this research evaluates the model's performance in classifying positive and negative sentiments, and compares the results with previous methods to determine the effectiveness of the approach used. One of the main challenges in social media sentiment analysis is the presence of informal language, such as slang words, which can affect the accuracy of the model. Therefore, this research also considers slang words normalization strategies to improve the model's understanding of the language variations used by social media users. In terms of contribution, this research provides insight into the effectiveness of SVMs in text-based sentiment analysis, particularly in handling non-standardized language. In addition, it offers a method that can improve accuracy in the classification of negative sentiment, which is often more difficult to recognize than positive sentiment. With this approach, this research not only contributes to improving the quality of sentiment analysis, but also provides a model improvement strategy that can form the basis for further studies in the fields of NLP and artificial intelligence (AI).

LITERATURE REVIEW

Sentiment analysis, also known as opinion mining, is a method used to identify and categorize sentiments in text as positive, negative, or neutral, aiming to analyze opinions, attitudes, and emotions related to various topics, products, or services. It is particularly useful for understanding how internet users perceive brands based on their comments and experiences. However, sentiment expression is complex, as users convey their feelings in nuanced ways, necessitating advanced methods for accurate interpretation. Text mining plays a crucial role in sentiment analysis by enabling classification, clustering, and extraction of relevant information from unstructured text, using processes such as text categorization, sentiment analysis, and entity-relationship modeling. Feature extraction is essential for effective sentiment classification, with the study employing TF-IDF (Term Frequency-Inverse Document Frequency) to measure word relevance within a document collection. The Naïve Bayes algorithm was

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14513
e-ISSN : 2541-2019
p-ISSN : 2541-044X

used for sentiment classification, demonstrating strong accuracy, precision, and recall in analyzing user comments. The dataset consisted of manually labeled comments from the YouTube trailer of *Money Heist* Season 4, categorized as positive, negative, or neutral, providing a structured foundation for model training. This literature survey highlights the core concepts and methodologies underpinning sentiment analysis, emphasizing the significance of text mining and machine learning techniques in understanding audience sentiments (Novendri et al., 2020).

Research conducted by (Ganie, 2023) focuses on the impact of informal language, such as emoticons, slang, and sarcasm, on the performance of sentiment analysis models, particularly in social media contexts. Sentiment analysis, or opinion mining, has become increasingly important across various fields, including e-commerce, marketing, and social media, as it helps analyze attitudes and emotions expressed in text data. A major challenge in this domain is the prevalence of informal language, which includes emoticons and slang, both of which can significantly affect model performance. Emoticons, or emojis, are widely used in online communication to express emotions, and their integration into sentiment analysis models has been shown to enhance accuracy, with some studies proposing frameworks that incorporate emoticon data with word embeddings. Similarly, slang influences sentiment classification outcomes, with research suggesting that including slang and emoticons in models can lead to improved accuracy. Sarcasm poses an even greater challenge, as it can mislead sentiment analysis models, necessitating effective sarcasm detection techniques. Despite growing research on informal language in sentiment analysis, there is still a lack of comprehensive studies examining the combined effects of emoticons, slang, and sarcasm, highlighting an opportunity for further research. This paper contributes to the field by investigating how informal language affects sentiment analysis models, concluding that while informal language has a limited impact, incorporating emoticon data can slightly improve model accuracy. In summary, the literature survey underscores the importance of understanding informal language in sentiment analysis, the challenges it presents, and the potential for enhancing model performance through the integration of emoticons and slang.

Sentiment analysis employing Support Vector Machine (SVM) techniques has gained traction due to its efficacy in classifying sentiments from diverse datasets. Obiedat et al. highlight a hybrid SVM-Particle Swarm Optimization model designed to tackle imbalanced datasets, demonstrating that optimizing such data substantially elevates classification performance in sentiment analysis of customer reviews (Obiedat et al., 2022). Furthermore, Azpiranda et al. corroborate the superiority of SVM over Naïve Bayes by reporting an accuracy of 82.48% for SVM, illustrating its strength in analyzing customer sentiments (Azpiranda et al., 2021). Notably, Singgalen indicates that SVM excels in accuracy and predictive capabilities, especially when combined with preprocessing techniques like SMOTE, which addresses class imbalances in sentiment classification (Singgalen, 2024). These findings collectively reinforce SVM pivotal role in sentiment analysis across various applications, including customer reviews and political sentiments (Anreaja et al., 2022; Kisma et al., 2024).

SMOTE-ENN (Synthetic Minority Over-sampling Technique with Edited Nearest Neighbor) is a potent method utilized in sentiment analysis to enhance classification accuracy, particularly when dealing with imbalanced datasets. Aljohani research emphasizes that SMOTE-ENN significantly improves performance metrics such as the F1-score, precision, and recall, which are crucial for maintaining information credibility in fake news detection systems in Arabic contexts (Aljohani, 2024). Similarly, Bounab et al. demonstrate the technique effectiveness in enhancing Medicare fraud detection, which parallels sentiment analysis tasks by addressing class imbalances through synthetic instance generation followed by a cleaning process using ENN (Bounab et al., 2024). Idakwo et al. illustrate the application of SMOTE-ENN in the context of chemical classification on highly imbalanced Tox21 datasets, highlighting its role in enhancing predictive accuracy by creating synthetic samples while correcting mislabeled instances (Idakwo et al., 2020). Furthermore, Al-Khazaleh et al. report that employing SMOTE-based techniques, including SMOTE-ENN, achieves remarkable results in sentiment classification of imbalanced Arabic data, presenting F1-scores indicative of high classification performance (Al-Khazaleh et al., 2024). These references collectively underscore the critical utility of SMOTE-ENN in sentiment analysis across various domains, particularly when faced with data imbalance issues.

METHOD

Before presenting the research method in Figure 1, it is essential to outline the systematic approach employed to gather and process the data. The method begins with the collection of movie reviews through the YouTube API, followed by cleaning the text data to ensure it is free from inconsistencies such as punctuation, slang, and irrelevant content. Text normalization techniques, including slang word normalization and tokenization, are then applied to standardize the text for further analysis. The data is further refined by removing stopwords and stemming words to their root form. Feature extraction using TF-IDF helps transform the textual data into numerical representations, enabling sentiment analysis with the VADER tool. After balancing the dataset and splitting it into training and testing subsets, hyperparameter tuning is performed to optimize model performance, which is then evaluated using various metrics such as accuracy, precision, recall, and F1-score. This structured approach ensures the integrity of the data and the reliability of the model's results.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14513
e-ISSN : 2541-2019
p-ISSN : 2541-044X

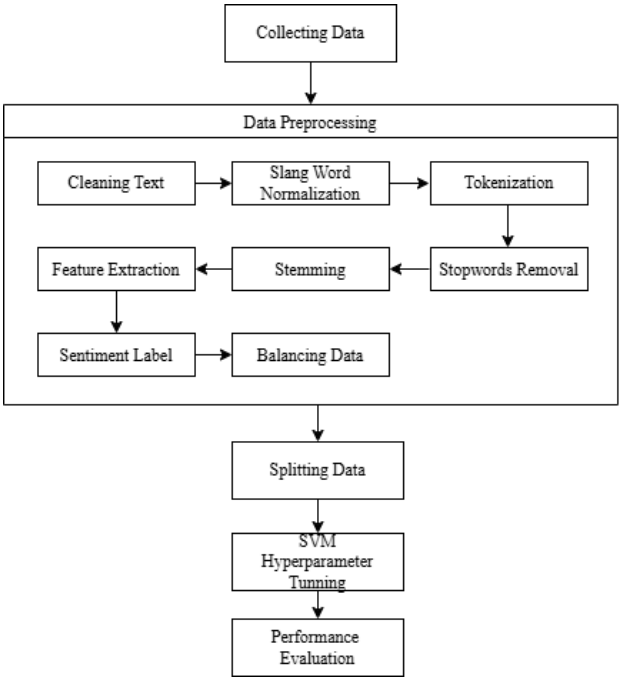

Fig 1. Research Method

1. Collecting Data
   The process of gathering review data for the movie *Pengabdi Setan 2: Communion* using the YouTube API crawling method involves utilizing the YouTube Data API service. This begins with enabling the YouTube API through Google Cloud Platform (GCP) and downloading the API credentials in JSON format to be applied in the crawling script.
2. Cleaning text
   The text cleaning process involves multiple steps to ensure the data is more refined and consistent. First, the text is converted to lowercase to eliminate distinctions between uppercase and lowercase letters. Then, punctuation and numbers are removed to prevent any disruptions in the analysis. Repeated characters appearing more than twice are reduced to enhance readability. Additionally, lines containing links are eliminated to remove irrelevant information. Emojis are also removed to further clean the data.
3. Slang Word Normalization
   Slang Word Normalization is the process of converting informal or non-standard words into their standard equivalents within a language. In text processing, this technique is commonly used to enhance the accuracy of text analysis, particularly in tasks such as sentiment analysis, text classification, and NLP-based machine learning. The process involves comparing words in the text against a slang dictionary that maps slang terms to their standard counterparts. In this implementation, the slang word pairs and their standard forms are sourced from a reference file, *kamusalay.csv*, which contains a comprehensive list of slang terms and their formal equivalents. This file is read and processed using *pandas* to transform it into a Python *dictionary*. With this dictionary representation, words in the text can be efficiently matched with their standardized versions. If a word appears in the slang dictionary, it is replaced with its formal equivalent, while words not found in the dictionary remain unchanged.
4. Tokenization
   Tokenization is the process of breaking down text into smaller units known as tokens. In NLP, these tokens can be words, phrases, or individual characters, depending on the specific analysis requirements.
5. Stopwords Removal
   Stopwords removal is the process of eliminating words that carry little to no significance in text analysis. These words are typically common and do not add much to contextual understanding, such as conjunctions or pronouns. The process involves comparing each word in the text against a predefined stopword list and removing any words that match.
6. Stemming
   Stemming is the process of reducing words to their root form by stripping affixes such as prefixes, infixes, and suffixes. The primary goal of stemming is to standardize word variations, making text analysis more efficient

*name of corresponding author

e-ISSN : 2541-2019
p-ISSN : 2541-044X

and enhancing the performance of NLP models. This process involves analyzing each word based on the morphological rules of the language and converting it into its base form.

7. Feature Extraction
   Feature extraction involves transforming text data into numerical representations to facilitate processing by modeling algorithms. One common approach is TF-IDF (Term Frequency-Inverse Document Frequency), which determines a word's importance within a document by considering both its frequency and its distribution across a set of documents. TF-IDF is a technique widely used in Natural Language Processing (NLP) to convert text data into numerical representations, enabling better processing by machine learning algorithms. It combines two key components: Term Frequency (TF), which measures how frequently a term appears in a document, and Inverse Document Frequency (IDF), which assesses the uniqueness of the term across the entire corpus. The TF value is calculated by dividing the frequency of the term in the document by the total number of terms in that document:

$$TF(t,d) = \frac{Total\ number\ of\ terms\ in\ document\ d}{Number\ of\ times\ term\ t\ appears\ in\ document\ d} \quad 1)$$

   IDF is calculated using the formula:

$$IDF(t,D) = log\left(\frac{Number\ of\ documents\ containing\ term\ t}{|D|}\right) \quad 2)$$

   Where |D| is the total number of documents in the corpus. The final TF-IDF score is obtained by multiplying the TF and IDF values, which reflects the importance of the term in a specific document while minimizing the weight of common, less informative words:

$$TF - IDF(t,d,D) = TF(t,d) \times IDF(t,D) \quad 3)$$

   This technique helps highlight important words, improve information retrieval, and select relevant features for further analysis, such as clustering or classification. To apply TF-IDF, the text must first be preprocessed (e.g., tokenization, stop word removal, stemming), then TF and IDF values are computed, and multiplied together to obtain the TF-IDF scores.
8. Sentiment Label
   Sentiment labeling is a technique for categorizing text based on its emotional polarity, such as positive or negative sentiment. In this study, sentiment analysis is performed using VADER (Valence Aware Dictionary and Sentiment Reasoner), a lexicon-based method specifically designed for analyzing short texts. VADER identifies the sentiment polarity of words and evaluates their emotional intensity by taking into account linguistic factors like punctuation, emphasis, and assertive words. The sentiment classification process is carried out by computing a compound score, which represents the overall sentiment orientation of the text. This score ranges from -1 to 1, where positive values indicate positive sentiment and negative values indicate negative sentiment. VADER is chosen for its efficiency and accuracy, particularly in processing informal language, such as social media comments.
9. Balancing Data
   Data balancing is a technique used to address class distribution imbalances in a dataset, particularly in classification tasks. In this study, the SMOTEENN (Synthetic Minority Over-sampling Technique and Edited Nearest Neighbors) method is utilized, combining both oversampling and undersampling approaches to equalize the number of samples between majority and minority classes. SMOTE (Synthetic Minority Over-sampling Technique) generates synthetic samples for the minority class by interpolating existing data points, while ENN (Edited Nearest Neighbors) removes samples from the majority class that are prone to misclassification. By integrating these methods, SMOTEENN not only increases the representation of the minority class but also eliminates data that might contribute to misclassification. This technique is applied to the training dataset to create a more balanced learning environment, helping the model reduce bias toward the majority class and enhancing its ability to accurately recognize patterns from both classes.
10. Splitting Data
    Data splitting is the process of partitioning a dataset into two primary subsets: training data (training set) and test data (testing set). This ensures that the model can learn from existing patterns and be evaluated using unseen data. In this study, the dataset is divided using the train-test split method with an 80:20 ratio, where 80% is allocated for training the model, and the remaining 20% is reserved for testing. To maintain a balanced class distribution in both subsets, the stratified sampling technique is employed, preventing the model from

*name of corresponding author

being biased toward the majority class. This approach helps mitigate overfitting, where the model becomes too specialized in the training data and struggles to generalize effectively to new data.

11. SVM Hyperparameter Tuning

In this study, the SVM hyperparameter tuning is performed using the GridSearchCV method from the scikit-learn library to find the optimal combination of hyperparameters that enhance the performance of the Support Vector Machine (SVM) model. The tuning process focuses on three key hyperparameters: C (the regularization parameter), kernel (the function used to map features into a higher-dimensional space), and gamma (a parameter that influences the shape of the kernel function for radial basis function (RBF) and polynomial kernels). The performance evaluation is based on accuracy, which represents the proportion of correct predictions made by the model. In mathematical terms, SVM aims to find the best hyperplane that separates two classes of data within the feature space. This can be described by the equation for linear classification:

$$f(x) = wTx + b \qquad 4)$$

After the search is finished, the optimal combination of hyperparameters that yields the highest accuracy score is chosen. These top results are then used to assess the model on the training data using other classification metrics like precision, recall, and F1-score, which offer a more comprehensive view of the model's performance across different aspects. The best model identified through this grid search is anticipated to enhance prediction accuracy and deliver better generalization on unseen data.

12. Performance Evaluation

Performance evaluation is the process of assessing a model's effectiveness in making accurate predictions. This is done using various metrics such as accuracy, precision, recall, and F1-score, each providing insights into the model's predictive balance between positive and negative classifications.

Accuracy is the overall proportion of correct predictions (both positive and negative) out of the total number of predictions.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad 5)$$

Precision is the proportion of positive predictions that are actually correct. It answers the question: "Of all instances predicted as positive, how many are actually positive?"

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

Recall is the proportion of actual positive cases that are correctly identified by the model. It answers the question: "Of all actual positive instances, how many were correctly predicted?"

$$Recall = \frac{TP}{TP+FN} \qquad 7)$$

The F1-score is the harmonic mean of precision and recall, providing a single metric that balances the two. It is useful when need a balance between precision and recall.

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad 8)$$

Additionally, the Confusion Matrix, a table displaying correct and incorrect predictions for each class, helps detect potential model bias. This evaluation is crucial for understanding how well the model generalizes to unseen data and identifying areas for improvement.

RESULT

1. Collecting Data

This study utilizes 3,375 YouTube comments on the movie *Pengabdi Setan 2: Communion*, encompassing publication time, username, comment text, and number of likes, as presented in Table 1. The comments exhibit diverse expressions, including informal language, slang, and emojis, capturing the spontaneous reactions of viewers. Since there are no missing values, all entries were suitable for sentiment analysis, which seeks to interpret audience responses to the film. A preprocessing phase was conducted to remove irrelevant elements such as URLs, excessive punctuation, and spam. Originally collected in JSON format via the YouTube API, the data was later converted into CSV format for ease of analysis. Various NLP techniques, including

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14513
e-ISSN : 2541-2019
p-ISSN : 2541-044X

tokenization, stemming, and sentiment classification using machine learning or deep learning, were applied to uncover patterns in audience opinions and the overall reception of the film.

Table 1. Youtube comments data

	publishedAt	authorDisplayName	textDisplay	likeCount
0	2024-11-25T00:50:22Z	@muhrois680	Pak ustadz nya mati mulu njirr kalah ama setan...	1
1	2024-11-23T06:52:38Z	@KharismaNurAzzahra-t6o	Gilaa ini kerennn bangetttt dlu udh pernah non...	0
2	2024-11-18T04:55:48Z	@caturwijinarko9892	<a href="https://www.youtube.com/watch?v=8LIHc...	0
3	2024-11-15T18:17:33Z	@wistinazz	TAKODT	0
4	2024-11-15T18:13:27Z	@wistinazz	takost	0
...	...	...	...	...
3370	2022-06-16T14:37:02Z	@yoansugiantara7673	Bakal keren banget pasti ini. Semua cast di fi...	117
3371	2022-08-01T09:23:40Z	@adiwidnyana1424	kyaknya jawaban atas teka teki di S1, belum ak...	2
3372	2022-08-22T18:03:51Z	@febifebrian1586	Dah selesai deh , soalnya kelar bapak nya send...	0
3373	2022-06-16T14:36:59Z	@RhayaFlicks	🔥🔥🔥🔥🔥	1
3374	2022-06-16T14:36:55Z	@mfauzan9686	Cant waittt	1

2. Cleaning Text

After text cleaning, the resulting data is cleaner and easier to analyze. All text has been converted to lowercase, punctuation marks and numbers have been removed, and characters that repeat more than twice have been simplified. Additionally, lines containing links or URLs have also been removed to ensure only relevant text remains. To further refine the text, emojis and special characters have been eliminated, preventing any unnecessary symbols from affecting the analysis. This step is crucial in maintaining textual consistency, especially when processing user-generated content that often includes emoticons or decorative symbols. The end result is a tidier and more structured set of text in Table 2, making it ready for further processing.

Table 2. Data after cleaning

	textDisplay
0	pak ustadz nya mati mulu njir kalah ama setan
1	gila ini keren banget dlu udh pernah nonton
2	takodt
3	takost
4	joko anwar adalah sebuah jaminan kualitas ting...
...	...
3278	bakal keren banget pasti ini semua cast di fil...
3279	kyaknya jawaban atas teka teki di s belum akan...
3280	dah selesai deh soalnya kelar bapak nya sendir...
3281	

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14513
e-ISSN : 2541-2019
p-ISSN : 2541-044X

3282	cant wait

3. Slang Word Normalization
In this case, kamusalay.csv is used as a reference to replace slang words with their official equivalents. This file contains a list of slang words along with their standard versions, which are then converted into a dictionary in order to be used in the text normalization process. By applying the normalization function, each word in the text is compared to the dictionary.csv, so that the non-standard words can be replaced with more appropriate spellings.

4. Tokenization
The tokenization results indicate that each text in the textDisplay column has been broken down into smaller words or tokens. For example, a sentence like "ustadz nya mati mulu njir kalah ama setan" has been transformed into a list of words such as ['pak', 'ustadz', 'nya', 'mati', 'mulu', 'astaga', 'kalah']. Similarly, another example, "gila ini keren banget dlu udh nonton", was tokenized into ['gila', 'ini', 'keren', 'banget', 'dulu', 'sudah', 'pernah']. Data supporting these tokenization results can be found in Table 3.

Table 3. Tokenization

	textDisplay	**textDisplay_normalized**
0	ustadz nya mati mulu njir kalah ama setan	[pak, ustadz, nya, mati, mulu, astaga, kalah, ...
1	gila keren banget dlu udh nonton	[gila, ini, keren, banget, dulu, sudah, pernah...
2	takodt	[takut]
3	takost	[takut]
4	joko anwar jamin kualitas tingi dukung pernyat...	[joko, anwar, adalah, sebuah, jaminan, kualita...
...	...	...
3276	woah ga sabar banget bom	[woah, tidak, sabar, banget, bakalan, bom]
3277	the real master piece	[the, real, master, piece]
3278	keren banget cast film prekuelnya pertanyan ab...	[bakal, keren, banget, pasti, ini, semua, cast...
3279	kyaknya teka teki s sepenuh jawab s brfeling g...	[kayaknya, jawaban, atas, teka, teki, di, s, b...
3280	dah selesai deh kelar nya yg janji iblis sehin...	[sudah, selesai, deh, soalnya, kelar, bapak, n...

5. Stopwords Removal
After the stopwords removal process, the text becomes more concise, emphasizing key words. For instance, the sentence "ustadz nya mati mulu njir kalah ama setan" is transformed into ['ustadz', 'nya', 'mati', 'mulu', 'astaga', 'kalah', 'setan'], eliminating words that do not significantly contribute to the analysis. Similarly, the sentence "gila keren banget dlu udh nonton" is simplified to ['gila', 'keren', 'banget', 'nonton'], making it more focused on the core meaning. Data illustrating these stopword removal results can be found in Table 4.

Table 4. Stopwords Removal

	textDisplay	**textDisplay_normalized**
0	ustadz nya mati mulu njir kalah ama setan	[ustadz, nya, mati, mulu, astaga, kalah, setan]
1	gila keren banget dlu udh nonton	[gila, keren, banget, nonton]
2	takodt	[takut]
3	takost	[takut]
4	joko anwar jamin kualitas tingi dukung pernyat...	[joko, anwar, jaminan, kualitas, tingi, menduk...
...	...	...
3276	woah ga sabar banget bom	[woah, sabar, banget, bom]
3277	the real master piece	[the, real, master, piece]
3278	keren banget cast film prekuelnya pertanyan ab...	[keren, banget, cast, film, prekuelnya, pertan...
3279	kyaknya teka teki s sepenuh jawab s brfeling g...	[kayaknya, teka, teki, s, sepenuhnya, terjawab...
3280	dah selesai deh kelar nya yg janji iblis sehin...	[selesai, deh, kelar, nya, perjanjian, iblis, ...

6. Stemming
After applying the stemming process with Literature, the words in the text are returned to their basic form. For example, the sentence “ustadz nya mati mulu njir kalah ama setan” is normalized to “ustadz nya mati mulu astaga kalah setan,” so that the text becomes more concise and easy to process further. In addition, words that had typos or variations, such as “takodt” and “takost,” were successfully corrected to “ takut,” which improved the overall quality of the text data. The results of this normalization can be seen in Table 5, which shows the comparison between the text before and after processing.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14513
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Table 5. Stemming

	textDisplay	textDisplay_normalized
0	ustadz nya mati mulu njir kalah ama setan	ustadz nya mati mulu astaga kalah setan
1	gila keren banget dlu udh nonton	gila keren banget nonton
2	takodt	takut
3	takost	takut
4	joko anwar jamin kualitas tingi dukung pernyat...	joko anwar jamin kualitas tingi dukung pernyat...
...	...	...
3276	woah ga sabar banget bom	woah sabar banget bom
3277	the real master piece	the real master piece
3278	keren banget cast film prekuelnya pertanyan ab...	keren banget cast film prekuelnya pertanyan ab...
3279	kyaknya teka teki s sepenuh jawab s brfeling g...	kayak teka teki s sepenuh jawab s brfeling lan...
3280	dah selesai deh kelar nya yg janji iblis sehin...	selesai deh kelar nya janji iblis sehinga ayah...

7. Feature Extraction

After performing feature extraction using TF-IDF (Term Frequency-Inverse Document Frequency), each word in the text is transformed into a numerical value that reflects its significance within the entire corpus. This method helps identify words that frequently appear in a specific document but are rare across the dataset, assigning higher weights to more meaningful terms. The extraction process results in a matrix where each row corresponds to a document (comment), each column represents a unique word from the corpus, and the matrix values indicate the TF-IDF scores. These scores determine how important a word is within a document relative to the whole dataset. A high TF-IDF score suggests that a word is particularly relevant and carries more weight in text analysis. This representation enables text data to be utilized in various NLP tasks, such as sentiment analysis, text classification, and information retrieval. Compared to the Bag of Words (BoW) approach, which only counts word occurrences, TF-IDF is more effective in addressing the issue of frequently appearing words that lack meaningful contribution to the analysis. The result can be seen in Table 6.

Table 6. Feature Extraction

	abad	abadi	abal	abang	abdi	...	zombiezombiean	zona	zonk	zumi
0	0	0	0	0	0	...	0	0	0	0
1	0	0	0	0	0	...	0	0	0	0
2	0	0	0	0	0	...	0	0	0	0
3	0	0	0	0	0	...	0	0	0	0
4	0	0	0	0	0	...	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...
3070	0	0	0	0	0	...	0	0	0	0
3071	0	0	0	0	0	...	0	0	0	0
3072	0	0	0	0	0.20845	...	0	0	0	0
3073	0	0	0	0	0	...	0	0	0	0
3074	0	0	0	0	0	...	0	0	0	0

8. Sentiment Label

After performing sentiment analysis using the VADER (Valence Aware Dictionary and Sentiment Reasoner) method, each text is categorized into two main groups, namely positive and negative, based on the compound score value. If the compound score value is positive or zero, the text is classified as positive, while if it is negative, the text is categorized as negative. The analysis results show that most of the texts have positive sentiments, such as “gila keren banget nonton” and “joko anwar jamin kualitas tinggi dukung pernyataan.” However, it is possible that some texts that actually have negative meanings are classified as positive, such as the word “takut,” because VADER is more optimal in analyzing English texts than Indonesian. Therefore, although this method is fast and efficient in sentiment classification, further validation or combination with other methods is needed to improve accuracy in Indonesian sentiment analysis. The results of this sentiment classification can be seen in Table 7, which displays the distribution of texts based on the resulting sentiment categories.

Table 7. Sentiment Label using VADER

	textDisplay	sentimen
0	ustadz nya mati mulu astaga kalah setan	positif
1	gila keren banget nonton	positif

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14513
e-ISSN : 2541-2019
p-ISSN : 2541-044X

2	takut	positif
3	joko anwar jamin kualitas tingi dukung pernyat...	positif
4	rela karakter si bikin meningal gitu sayang an...	positif

9. Balancing Data

After applying SMOTEENN (Synthetic Minority Over-sampling Technique and Edited Nearest Neighbors) to balance the data, the previously highly imbalanced class distribution changed significantly, as illustrated in Table 8. Initially, the dataset contained a much larger number of positive sentiment samples (2,968) compared to negative sentiment samples (only 107), which could lead to model bias toward the majority class and a reduced ability to identify patterns from the minority class. However, after implementing SMOTEENN on the training set, the data distribution became more balanced, with 2,942 negative samples and 2,968 positive samples. This method combines SMOTE (oversampling) to generate synthetic samples for the minority class and ENN (undersampling) to eliminate potential noise samples. As a result, the training data becomes more representative, improving the model's ability to classify sentiment more accurately. This distribution shift, as presented in Table 8, ensures that the model does not disproportionately favor the majority class and can better handle negative sentiment data.

Table 8. Balancing Data using SMOTEENN

Sentimen	Before SMOTEENN	After SMOTEENN
Positif	2968	2968
Negatif	107	2942

10. Splitting Data

The data splitting process follows an 80:20 ratio, where 80% of the data is allocated for training the model and 20% is reserved for testing. As detailed in Tabel 9, the training set contains 4728 data points, while the testing set holds 1182 data points. Additionally, the stratified splitting method ensures that the class distribution is maintained in both sets, providing a balanced representation of the dataset for the model to learn from.

Table 9. Splitting Data

Dataset	Count	Percentage
Training Data	4728	80%
Testing Data	1182	20%

11. SVM Hyperparameter Tuning

During the hyperparameter tuning stage, the Support Vector Machine (SVM) model is optimized using the Grid Search technique to identify the best parameter combination that yields the highest accuracy. The parameters tested include C (regularization parameter) with values of 0.1, 1, and 10, the kernel type (linear, rbf, and poly), and gamma, which influences the rbf and poly kernels, with scale and auto options. This tuning process is conducted using cross-validation (cv=5) to ensure that the evaluation results are more generalizable and do not lead to overfitting. After executing Grid Search, the optimal parameter combination was found to be C=10, kernel=linear, and gamma=scale.

12. Performance Evaluation

In the evaluation of the best model performed on the test data, the results show excellent performance with precision, recall, and f1-score values reaching 1.00 for both classes (0 and 1) respectively. This indicates that the model successfully classified each instance perfectly, both for the negative (0) and positive (1) classes, without any misclassification. Based on Table 10 of the Evaluation Matrix, the model's accuracy on the test data reached 100%, which indicates that the model did not make any errors in predicting the test data. Furthermore, in Figure 2 Confusion Matrix, it can be seen that the model generates its confusion matrix with perfect values, namely 594 for correct negative predictions and 588 for correct positive predictions, without any prediction errors (false positives or false negatives). This further confirms that the model is highly accurate and able to distinguish between the two classes very well.

Table 10. Matrix Evaluation

	Precision	Recall	F1-score	Support
Negative	100%	100%	100%	594
Positive	100%	100%	100%	588
Macro Average	100%	100%	100%	615
Weighted Average	100%	100%	100%	615

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14513
e-ISSN : 2541-2019
p-ISSN : 2541-044X

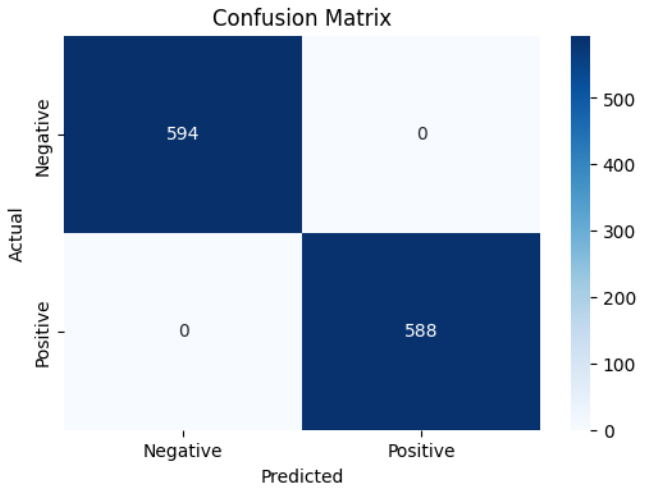

Fig 2. Confusion Matrix

DISCUSSIONS

The sentiment analysis of YouTube comments on *Pengabdi Setan 2: Communion* highlights the significant impact of pre-processing techniques, such as slang word normalization, text cleaning, and tokenization, in improving model performance. The dataset, which included a mix of formal and informal language, slang, emojis, and punctuation, was made more uniform and suitable for analysis through cleaning processes like removing URLs, special characters, and excessive punctuation. Slang word normalization, using a dictionary-based approach, ensured consistency by replacing informal terms with formal equivalents, such as converting "takodt" and "takost" to "takut." Tokenization and stopword removal further refined the dataset, enabling the model to focus on meaningful units and eliminating unnecessary words. Stemming reduced words to their base form, improving accuracy. Feature extraction through TF-IDF allowed the model to capture significant terms, while sentiment classification using VADER was effective but occasionally misclassified informal language. To address class imbalance, SMOTEENN was used, balancing the sentiment distribution and enhancing model accuracy. The data split into an 80:20 ratio ensured robust training and testing, with the final SVM model achieving perfect precision, recall, and F1-score values, resulting in 100% accuracy. This study demonstrates the effectiveness of combining text pre-processing techniques and SVM for sentiment analysis, overcoming challenges posed by informal language and achieving reliable insights from social media data.

CONCLUSION

In conclusion, this study successfully demonstrates that combining advanced text pre-processing techniques, such as slang word normalization, tokenization, and stemming, with machine learning models like Support Vector Machine (SVM) can significantly improve sentiment analysis accuracy on user-generated content, such as YouTube comments. The preprocessing steps were essential in handling informal language, slang, and noisy text, ensuring that the data was more consistent and meaningful for analysis. By addressing class imbalance through SMOTEENN and optimizing the SVM model with hyperparameter tuning, the model achieved perfect classification performance with 100% accuracy. Although VADER showed promising results in sentiment classification, the study also highlighted its limitations with informal language, suggesting the need for further adaptation to handle slang more effectively. Overall, the combination of these techniques provides a robust framework for sentiment analysis, offering valuable insights into the audience's perception of media content and paving the way for more accurate social media analytics.

REFERENCES

Al-Khazaleh, M. J., Alian, M., & Jaradat, M. A. (2024). Sentiment analysis of imbalanced Arabic data using sampling techniques and classification algorithms. *Bulletin of Electrical Engineering and Informatics*, *13*(1), 607–618. https://doi.org/10.11591/eei.v13i1.5886

Aljohani, E. (2024). Enhancing Arabic Fake News Detection: Evaluating Data Balancing Techniques Across Multiple Machine Learning Models. *Engineering, Technology and Applied Science Research*, *14*(4), 15947–15956. https://doi.org/10.48084/etasr.8019

Anreaja, L. J., Harefa, N. N., Negara, J. G. P., Pribyantara, V. N. H., & Prasetyo, A. B. (2022). Naive Bayes and

*name of corresponding author

Support Vector Machine Algorithm for Sentiment Analysis Opensea Mobile Application Users in Indonesia. *JISA(Jurnal Informatika Dan Sains)*, *5*(1), 62–68. https://doi.org/10.31326/jisa.v5i1.1267

Azpiranda, N., Supianto, A. A., Setiawan, N. Y., Suryawati, E., Yuwana, R. S., & Febriandirza, A. (2021). Sentiment Anlysis On Customer Reviews Using Support Vector Machine and Usability Scoring Using System Usability Scale. *Journal of Information Technology and Computer Science*, *6*(3), 236–251. https://doi.org/10.25126/jitecs.202163330

Bagate, R. A., & Suguna, R. (2021). Sarcasm detection of tweets without #sarcasm: Data science approach. *Indonesian Journal of Electrical Engineering and Computer Science*, *23*(2), 993–1001. https://doi.org/10.11591/ijeecs.v23.i2.pp993-1001

Bounab, R., Zarour, K., Guelib, B., & Khlifa, N. (2024). Enhancing Medicare Fraud Detection Through Machine Learning: Addressing Class Imbalance With SMOTE-ENN. *IEEE Access*, *12*(March), 54382–54396. https://doi.org/10.1109/ACCESS.2024.3385781

Chamekh, A., Mahfoudh, M., & Forestier, G. (2022). Sentiment Analysis Based on Deep Learning in E-Commerce. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *13369 LNAI*, 498–507. https://doi.org/10.1007/978-3-031-10986-7_40

Chinedu, E. Q., Asogwa, E. C., Sunday, B. T., & Macdonald, N. (2023). Unraveling Emotions: Contemporary Approaches in Sentiment Analysis. *Journal of Sensor Networks and Data Communications*, *3*(1), 223–230. https://doi.org/10.33140/jsndc.03.01.14

Ganie, A. G. (2023). Presence of informal language, such as emoticons, hashtags, and slang, impact the performance of sentiment analysis models on social media text? *ArXiv Preprint ArXiv:2301.12303*.

Idakwo, G., Thangapandian, S., Luttrell, J., Li, Y., Wang, N., Zhou, Z., Hong, H., Yang, B., Zhang, C., & Gong, P. (2020). Structure–activity relationship-based chemical classification of highly imbalanced Tox21 datasets. *Journal of Cheminformatics*, *12*(1), 1–19. https://doi.org/10.1186/s13321-020-00468-x

Khan, M. Y., Ahmed, T., Siddiqui, M. S., & Wasi, S. (2023). Cognitive Relationship-Based Approach for Urdu Sarcasm and Sentiment Classification. *IEEE Access*, *11*(September), 126661–126690. https://doi.org/10.1109/ACCESS.2023.3325048

Kisma, A. J. N., Arsi, P., & Subarkah, P. (2024). Sentiment Analysis Regarding Candidate Presidential 2024 Using Support Vector Machine Backpropagation Based. *JTAM (Jurnal Teori Dan Aplikasi Matematika)*, *8*(1), 96. https://doi.org/10.31764/jtam.v8i1.17294

Kularbphettong, K., Roonrakwit, P., & Boonseng, C. (2024). Sentiment analysis of the awareness of environmental sustainability. *Edelweiss Applied Science and Technology*, *8*(3), 145–155. https://doi.org/10.55214/25768484.v8i3.847

Kumar, A., Narapareddy, V. T., Srikanth, V. A., Malapati, A., & Neti, L. B. M. (2020). Sarcasm Detection Using Multi-Head Attention Based Bidirectional LSTM. *IEEE Access*, *8*, 6388–6397. https://doi.org/10.1109/ACCESS.2019.2963630

Lestari, U., & Anugrahni, D. (2021). Sentiment Analysis of Performance Effectiveness of Malioboro Pedestrian Using Sentistrength Method on Twitter. *Jurnal TAM (Technology Acceptance Model)*, *12*(1), 75. https://doi.org/10.56327/jurnaltam.v12i1.1044

Munggaran, J. P., Alhafidz, A. A., Taqy, M., Agustini, D. A. R., & Munawir, M. (2023). Sentiment Analysis of Twitter Users' Opinion Data Regarding the Use of ChatGPT in Education. *Journal of Computer Engineering, Electronics and Information Technology*, *2*(2), 75–88. https://doi.org/10.17509/coelite.v2i2.59645

Novendri, R., Callista, A. S., Pratama, D. N., & Puspita, C. E. (2020). Sentiment Analysis of YouTube Movie Trailer Comments Using Naïve Bayes. *Bulletin of Computer Science and Electrical Engineering*, *1*(1), 26–32. https://doi.org/10.25008/bcsee.v1i1.5

Nurodin, M. I., & Puspitarani, Y. (2023). Phrase Detection'S Impact on Sentiment Analysis of Public Opinion and Online Media Toward Political Figures. *Jurnal Riset Informatika*, *6*(1), 67–76. https://doi.org/10.34288/jri.v6i1.XXX

Obiedat, R., Qaddoura, R., Al-Zoubi, A. M., Al-Qaisi, L., Harfoushi, O., Alrefai, M., & Faris, H. (2022). Sentiment Analysis of Customers' Reviews Using a Hybrid Evolutionary SVM-Based Approach in an Imbalanced Data Distribution. *IEEE Access*, *10*, 22260–22273. https://doi.org/10.1109/ACCESS.2022.3149482

Omar, A., & Hamouda, W. I. (2021). A Sentiment Analysis of Egypt's New Real Estate Registration Law on Facebook. *International Journal of Advanced Computer Science and Applications*, *12*(4), 656–663. https://doi.org/10.14569/IJACSA.2021.0120481

Potamias, R. A., Siolas, G., & Stafylopatis, A. G. (2020). A transformer-based approach to irony and sarcasm detection. *Neural Computing and Applications*, *32*(23), 17309–17320. https://doi.org/10.1007/s00521-020-05102-3

Prestianta, A. M. (2021). Mapping the ASEAN YouTube Uploaders. *Jurnal ASPIKOM*, *6*(1), 1.

*name of corresponding author

e-ISSN : 2541-2019
p-ISSN : 2541-044X

https://doi.org/10.24329/aspikom.v6i1.761

Rao, S. A., Ravi, M. S., Zhao, J. W., Sturgeon, C., & Bilimoria, K. Y. (2020). Social Media Responses to Elective Surgery Cancellations in the Wake of COVID-19. *Annals of Surgery*, *272*(3), E246–E248. https://doi.org/10.1097/SLA.0000000000004106

Redjeki, S., & Widyarto, S. (2022). Comparison of Seven Machine Learning Algorithms in the Classification of Public Opinion. *Tech-E*, *5*(2), 143–149. https://doi.org/10.31253/te.v5i1.1046

Singgalen, Y. A. (2024). Sentiment Classification of Robot Hotel Content using NBC and SVM Algorithm. *Journal of Computer System and Informatics (JoSYC)*, *5*(2), 442–453. https://doi.org/10.47065/josyc.v5i2.4924

Syafia, A. N., Hidayattullah, M. F., & Suteddy, W. (2023). Studi Komparasi Algoritma SVM Dan Random Forest Pada Analisis Sentimen Komentar Youtube BTS. *Jurnal Informatika: Jurnal Pengembangan IT*, *8*(3), 207–212. https://doi.org/10.30591/jpit.v8i3.5064

Williams, L., Anthi, E., & Burnap, P. (2024). Comparing Hierarchical Approaches to Enhance Supervised Emotive Text Classification. *Big Data and Cognitive Computing*, *8*(4). https://doi.org/10.3390/bdcc8040038

Xiong, W., Zuo, Y., Zhang, M., Zhang, C., & Guo, C. (2024). Research on Sentiment Analysis of E-commerce Live Comments based on Text Mining. *Frontiers in Computing and Intelligent Systems*, *6*(3), 34–36. https://doi.org/10.54097/c2wofcb2

Zhan, T., Shi, C., Shi, Y., Li, H., & Lin, Y. (2024). Optimization techniques for sentiment analysis based on LLM (GPT-3). *Applied and Computational Engineering*, *67*(1), 27–33. https://doi.org/10.54254/2755-2721/67/2024ma0060

*name of corresponding author