Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, Number 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4670

Spam Detection on YouTube Comments Using Advanced Machine Learning Models: A Comparative Study

Gregorius Airlangga[1*]
[1]Atma Jaya Catholic University of Indonesia, Indonesia
[1]gregorius.airlangga@atmajaya.ac.id

open access

***Corresponding Author**

Article History:
Submitted: 16-09-2024
Accepted: 23-09-2024
Published: 04-10-2024

Keywords:
Spam Detection; Machine Learning; YouTube Comments; Text Classification; LinearSVC

ABSTRACT

The exponential growth of user-generated content on platforms like YouTube has led to an increase in spam comments, which negatively affect the user experience and content moderation efforts. This research presents a comprehensive comparative study of various machine learning models for detecting spam comments on YouTube. The study evaluates a range of traditional and ensemble models, including Linear Support Vector Classifier (LinearSVC), RandomForest, LightGBM, XGBoost, and a VotingClassifier, with the goal of identifying the most effective approach for automated spam detection. The dataset consists of labeled YouTube comments, and text preprocessing was performed using Term Frequency-Inverse Document Frequency (TF-IDF) vectorization. Each model was trained and evaluated using a stratified 10-fold cross-validation to ensure robustness and generalizability. LinearSVC outperformed all other models, achieving an accuracy of 95.33% and an F1-score of 95.32%. The model demonstrated superior precision (95.46%) and recall (95.33%), making it highly effective in distinguishing between spam and legitimate comments. The results highlight the potential of LinearSVC for real-time spam detection systems, offering a reliable balance between accuracy and computational efficiency. Furthermore, the study suggests that while ensemble models like RandomForest and VotingClassifier performed well, they did not surpass the simpler LinearSVC model in this context. Future work will explore the incorporation of deep learning techniques, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), to capture more complex patterns and further enhance spam detection accuracy on social media platforms like YouTube.

INTRODUCTION

The growth of social media platforms has fundamentally altered how individuals interact and share content globally (Appel et al., 2020; Evans et al., 2021; Manca et al., 2021). YouTube, one of the largest video-sharing platforms, enables users to engage with content creators and other viewers through comments, contributing to a dynamic online community(Hussain et al., 2024). However, this environment has become increasingly inundated with spam comments, ranging from irrelevant promotions to malicious links. This not only disrupts the user experience but also poses significant challenges to content moderation. The detection and removal of spam comments have thus become critical tasks to maintain the platform's integrity and ensure a safe environment for users (Jain et al., 2021; Rao et al., 2021; J. Wang et al., 2021). Manual moderation of millions of comments daily is unfeasible. Therefore, there is a pressing need for automated spam detection systems (Gongane et al., 2022). With the advances in machine learning and natural language processing (NLP), the development of models that can accurately classify comments as spam or legitimate has gained substantial attention [8]. However, existing methods face challenges in adapting to the evolving nature of spam tactics, which necessitates the exploration of more sophisticated models.

Spam detection has been a focal point of research in various domains, including email filtering, social media moderation, and online forums. Early methods relied on rule-based approaches that utilized predefined keywords and patterns to identify spam (Akinyelu, 2021; Jahan & Oussalah, 2023; Rastogi et al., 2020). While effective in the initial stages, these static systems quickly became inadequate as spammers developed more sophisticated evasion techniques (Salman et al., 2024). Machine learning emerged as a solution, introducing models capable of learning from labeled data and adapting to changing spam patterns. The Naive Bayes classifier was among the first to be employed for spam detection, particularly in email filtering systems (Jáñez-Martino et al., 2023). However, as the complexity of spam increased, more advanced algorithms such as Support Vector Machines (SVMs) and decision trees were adopted to improve classification accuracy. In the context of social media platforms, spam detection has primarily focused on two approaches: content-based and behavior-based methods (Rao et al., 2021). Content-based methods analyze the textual content of comments to identify spam, while behavior-based methods examine user activity patterns, including post frequency and network associations (Abkenar et al., 2023). For platforms like YouTube, which generate vast amounts of

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, Number 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4670

textual data, content-based methods are particularly relevant. Traditional text vectorization techniques like TfidfVectorizer have been widely used to convert textual data into numerical representations that machine learning models can process (Yang et al., 2022). More recently, deep learning models such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks have been applied to text classification tasks, demonstrating enhanced capabilities in capturing complex patterns within textual data (Hassani et al., 2020). However, these models require substantial computational resources and large datasets for effective training.

Despite the extensive research on spam detection for social media platforms like Twitter and Facebook, there remains a gap in the application of these methods to YouTube comments. The unique nature of YouTube, where textual comments are directly associated with video content, presents specific challenges that have not been thoroughly explored in existing studies (Teng et al., 2020). Furthermore, while some research has applied machine learning models to this domain, there is a need for a comprehensive evaluation of both traditional and ensemble models to identify the most effective approach for this specific context (Mohammed & Kora, 2023). The urgency of addressing spam detection on YouTube stems from the increasing prevalence and sophistication of spam tactics. Spam comments can mislead viewers, expose them to harmful content such as phishing links, and degrade the overall user experience (Abbasi et al., 2021). For content creators, the presence of spam can negatively impact engagement metrics, reducing visibility and potential revenue. Given YouTube's vast scale, processing millions of comments each day and manual moderation is not a viable solution (Galli et al., 2022). As spam evolves to become more adaptive and personalized, the need for automated detection systems becomes critical. Robust spam detection is essential not only for maintaining the platform's credibility but also for protecting users from potential harm (Alkhamees et al., 2021). The current state of spam detection has seen a shift toward the use of ensemble models and hybrid approaches, which combine multiple machine learning algorithms to enhance classification performance (Rao et al., 2023). Ensemble methods, such as Random Forest and Gradient Boosting, have demonstrated considerable success in text classification tasks, including spam detection (Zhang, 2024). By aggregating the predictions of several base classifiers, these models reduce the risk of misclassification and improve overall accuracy. Voting Classifiers, which combine the predictions of multiple distinct models, have also been effectively employed in spam detection tasks, offering a balanced approach that captures the strengths of individual classifiers (Gaafar et al., 2022).

Parallel to ensemble methods, deep learning models have gained traction in recent years, particularly for large-scale text classification tasks. CNNs and Recurrent Neural Networks (RNNs), including LSTMs, have shown promise in processing sequential data and identifying intricate patterns within text (Al Zoubi & others, 2024). However, their application in real-time spam detection is limited by their computational demands and latency issues. In feature engineering, TfidfVectorizer remains a popular tool for converting text into numerical data (M. Wang et al., 2020). However, more advanced techniques, such as word embeddings (e.g., Word2Vec, GloVe) and contextual embeddings (e.g., BERT, GPT), are increasingly being adopted for their ability to capture nuanced semantic relationships in text. Despite their advantages, these methods often require significant computational resources and are complex to implement without access to large-scale datasets (Akinyelu, 2021).

This research aims to develop a scalable and efficient spam detection system for YouTube comments using various machine learning models. By systematically comparing a range of machine learning algorithms including traditional models such as Naive Bayes and Logistic Regression, as well as more advanced ensemble models like Random Forest, Gradient Boosting, XGBoost, and LightGBM, the study seeks to identify the most effective approach for classifying comments on YouTube. Additionally, the research explores the potential of a Voting Classifier that combines multiple models to achieve superior detection performance. Although substantial progress has been made in spam detection across different online platforms, there is a notable gap in research specifically focusing on YouTube comments. Most existing studies have concentrated on other social media platforms, overlooking the unique challenges posed by YouTube's comment system. Moreover, while models such as Random Forest, SVM, and Naive Bayes have been employed for spam detection, their performance in the context of YouTube requires further investigation. Existing research also tends to overlook the importance of model interpretability, which is crucial for understanding decision-making processes in spam detection systems. Another challenge lies in the computational efficiency of these models; many advanced algorithms are resource-intensive and may not be suitable for real-time spam detection on a platform as vast as YouTube.

The rest of this paper is organized as follows: The literature review explains the previous and existing research, then, the methodology section outlines the dataset used, detailing the preprocessing steps and describing the machine learning models implemented for spam detection. The Results section presents the outcomes of the experimental evaluations, providing a thorough analysis of the models' performance using metrics such as accuracy, precision, recall, and F1-score, and in the discussion section, the implications of the findings are explored, particularly in the context of model complexity, computational efficiency, and interpretability. Finally, the Conclusion summarizes the key contributions of the research, highlighting its significance and proposing potential directions for future work. This structure ensures a systematic exploration of the research questions, grounded in both theoretical and practical insights.

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, Number 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4670

LITERATURE REVIEW

Spam detection has been a focal point of research across various domains, evolving from simple rule-based systems to complex machine learning models. This evolution reflects the increasing sophistication of spam tactics and the need for adaptable solutions. In the context of social media and video-sharing platforms like YouTube, the challenge of accurately identifying and filtering spam comments has grown more complex due to the diverse nature of content and user interactions. This section reviews the literature on spam detection, focusing on the progression of methodologies and highlighting the gaps that this research aims to address. The initial efforts in spam detection were largely centered around email filtering, where rule-based systems played a pivotal role. These systems utilized predefined keywords, blacklists, and regular expressions to identify unsolicited messages (Agarwal et al., 2024). While these approaches were effective against early forms of spam, they were inherently rigid and struggled to adapt to evolving spam strategies. Spammers quickly learned to bypass these static rules by using obfuscation techniques, such as misspelling words or varying their messages. The high maintenance costs and the need for constant updates limited the scalability of rule-based systems, particularly in dynamic environments like social media.

To overcome these limitations, statistical and probabilistic models were introduced. Naive Bayes classifiers became prominent in email spam filtering, leveraging word frequencies and conditional probabilities to classify messages (Andresini et al., 2022). Naive Bayes offered a more flexible and probabilistic approach to spam detection, capable of handling large vocabularies and providing insights into the likelihood of a message being spam. Despite its effectiveness, the Naive Bayes classifier assumes feature independence, which is not always valid in real-world data, resulting in suboptimal performance when dealing with more complex spam scenarios, particularly those involving contextual or sequential dependencies. As spam detection moved into the realm of social media, the complexity of the problem increased. Social media spam includes a variety of forms, such as fake profiles, phishing links, and promotional content. Unlike email, social media spam often involves a mix of text, images, and behavioral patterns, making detection more challenging. Researchers have explored both content-based and behavior-based methods for spam detection in this domain (Nascimento et al., 2023)

Content-based methods focus on analyzing the textual content of posts, comments, or messages. Techniques such as Term Frequency-Inverse Document Frequency (TF-IDF) and word embeddings (Khan et al., 2021) have been widely used to convert textual data into numerical features. These features are then fed into machine learning models like Support Vector Machines (SVMs) and decision trees to classify content as spam or legitimate (Hakak et al., 2021). SVMs have been favored for their ability to handle high-dimensional feature spaces and find optimal decision boundaries between classes. However, these models require careful feature engineering and preprocessing to achieve high accuracy, and their performance can be hindered by highly imbalanced datasets, a common issue in spam detection where spam comments often represent a small fraction of the total data. Behavior-based methods, on the other hand, analyze user activity patterns to identify spam. This approach involves examining metrics such as posting frequency, timing, and network associations. For instance, spammers often exhibit abnormal behavior, such as posting many comments within a short time frame or engaging with a disproportionately large number of users. Random Forest and other ensemble models have been applied to capture these behavioral patterns, showing improved detection rates (Barushka, 2020). However, behavior-based methods often require access to detailed user metadata, which may raise privacy concerns and is not always readily available for analysis.

The limitations of traditional machine learning models in handling the nuances of spam detection have led to the exploration of deep learning techniques. Deep learning models, particularly Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have demonstrated significant success in text classification tasks due to their ability to automatically learn complex patterns and semantic structures in text data (Gunturi & Sarkar, 2021). CNNs are effective in identifying local patterns within text, such as n-grams, while RNNs, especially Long Short-Term Memory (LSTM) networks, excel in capturing sequential dependencies, making them suitable for detecting context-sensitive spam. However, the application of deep learning models in spam detection is not without challenges. These models require large, labeled datasets for training, which can be a limiting factor in domains where labeled data is scarce or costly to obtain. Additionally, deep learning models are computationally intensive, requiring substantial processing power and memory. In real-time settings, such as YouTube's comment moderation, the latency introduced by deep learning models may render them impractical for deployment.

In parallel with deep learning, advanced text representation techniques have emerged to enhance spam detection performance. Word embeddings like Word2Vec (Gasparetto et al., 2022) and GloVe (Zeakis et al., 2023) capture the semantic relationships between words, providing richer representations compared to traditional vectorization methods like TF-IDF. More recently, contextual embeddings from transformer-based models like BERT (Reshma, 2020) have further improved the ability of models to understand nuanced meanings in text. Despite their advantages, these methods often demand extensive computational resources and are complex to integrate into existing workflows, particularly in large-scale environments like YouTube. Ensemble methods have gained prominence in spam detection due to their ability to combine the strengths of multiple base models. Random Forest, an ensemble of decision trees, has been widely adopted for spam detection tasks and has shown robust performance across different datasets (Liu et al., 2021). Gradient Boosting techniques, such as XGBoost (Bazzaz Abkenar et al., 2021) and (Fafalios et al., 2020), build models

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, Number 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4670

sequentially, focusing on correcting the errors of their predecessors to improve accuracy. These models are particularly effective in handling imbalanced datasets and capturing complex decision boundaries. Voting Classifiers represent another ensemble approach that aggregates the predictions of multiple models to achieve a more balanced and robust classification outcome. By combining the strengths of diverse algorithms, Voting Classifiers can mitigate the weaknesses of individual models and enhance overall performance. Hybrid models, which integrate machine learning and deep learning techniques, have also been explored to leverage the advantages of both approaches (Kavzoglu & Teke, 2022). These models typically use CNNs or RNNs for feature extraction, followed by ensemble classifiers for the final prediction, thereby achieving high accuracy while maintaining computational efficiency.

While significant advances have been made in spam detection across various platforms, the application of these techniques to YouTube comments presents unique challenges. YouTube's comment system is intrinsically linked to video content, introducing a multimodal aspect to spam detection that is not present in text-only platforms. Spam comments on YouTube can vary depending on the type of video, the target audience, and the spammer's objectives, requiring models to be highly adaptable to different contexts. Existing research on YouTube spam detection has often employed traditional machine learning models such as Naive Bayes, SVM, and Random Forest, with varying degrees of success (Ahmed et al., 2023; Saumya & Singh, 2022). However, these models may not fully capture the dynamic and evolving nature of spam tactics on YouTube. Additionally, the scale of YouTube's platform, with millions of comments generated daily, necessitates models that are not only accurate but also computationally efficient. Real-time spam detection requires models that can process large volumes of data swiftly, making computational efficiency a critical consideration. The need for real-time processing, coupled with the evolving nature of spam, underscores the importance of exploring advanced ensemble methods and hybrid models that can adapt to the dynamic environment of YouTube.

The literature highlights the evolution of spam detection methods, from early rule-based approaches to advanced machine learning and deep learning techniques. However, despite the progress, several gaps remain, particularly in the context of YouTube. Most notably, there is a need for a comprehensive evaluation of various machine learning models tailored specifically for YouTube's comment system, addressing the unique challenges of multimodal content and real-time processing. While deep learning and ensemble methods have shown promise, their applicability to YouTube spam detection has not been fully explored in a comparative context. This research aims to bridge these gaps by systematically evaluating a diverse set of machines learning models, including traditional classifiers, advanced ensemble methods, and a Voting Classifier designed to combine the strengths of multiple algorithms. By focusing on YouTube-specific challenges, such as the diversity of comment styles and the need for computational efficiency, this study seeks to identify the most effective spam detection approach for this platform. The findings will not only contribute to the literature on spam detection but also provide practical insights for enhancing automated moderation systems on YouTube and similar platforms.

METHOD

The methodology for this research involves a systematic approach encompassing dataset preparation, text preprocessing, feature extraction, model development, cross-validation, and model evaluation. This section provides a detailed explanation of the mathematical formulations, algorithms, and processes utilized in each stage.

Dataset Preparation

The dataset used in this study consists of YouTube comments, each labeled as either spam (1) or legitimate (0). The dataset includes features such as Author, Date, Content, VideoName and the target variable CLASS. The raw dataset is denoted as $(D = \{(X_i, y_i)\}_{i=1}^{N})$, where (X_i) represents the feature vector for the (i)-th comment, and $(y_i \in \{0,1\})$ is the corresponding label. Here, (N) is the total number of comments. To ensure data integrity, duplicate entries are removed using the transformation $D' = \{(X_i, y_i) \in D : X_i \neq X_j \quad \forall i \neq j\}$. The resulting dataset (D') forms the basis for further processing. The dataset is then explored for missing values, which can affect model performance. Let $(\delta(X_i))$ be an indicator function that checks for missing values in the feature vector $\delta(X_i) = 1\ if any X_{ij} = NaN, \forall j, 0\ otherwise$. Then, $if(\sum_{i=1}^{N} \delta(X_i))$ is non-zero, appropriate imputation techniques are applied, such as replacing missing values with the mean, median, or mode of the respective feature.

Text Preprocessing and Feature Engineering

The primary input feature for classification is the textual content of comments. To enhance the input features, text-based attributes Author, Video_Name and Content are concatenated into a single composite feature $\text{comment\_info} = \text{AUTHOR} + \text{VIDEO\_NAME} + \text{CONTENT}$. Let $(T = \{t_i\}_{i=1}^{N})$ represent the set of composite text features. Each element (t_i)undergoes the following preprocessing steps: tokenization, lowercasing, stopword removal, and stemming/lemmatization. Tokenization splits the text into individual tokens $t_i = [w_{i1}, w_{i2}, \ldots, w_{in}]$. Tokens are converted to lowercase to ensure case insensitivity, stopwords are removed to reduce noise, and stemming/lemmatization reduces tokens to their root forms.

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, Number 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4670

Feature Extraction Using TF-IDF Vectorization

The preprocessed text is transformed into numerical form using Term Frequency-Inverse Document Frequency (TF-IDF) vectorization. Let $(\mathcal{V} = \{v_k\}_{k=1}^{V})$ denote the vocabulary, where (V) is the vocabulary size. The TF-IDF score for each term (v_k) in a document (t_i) is calculated as $\text{tfidf}(t_i, v_k) = \text{tf}(t_i, v_k) \times \text{idf}(v_k)$, where $\text{tf}(t_i, v_k) = \frac{\text{count}(v_k, t_i)}{\sum_{v_j \in t_i} \text{count}(v_j, t_i)}$ is the term frequency, and $\text{idf}(v_k) = \log\left(\frac{N}{1+|\{t_i : v_k \in t_i\}|}\right)$ is the inverse document frequency. This process transforms each comment (t_i) into a vector in (R^V) is $v(t_i) = [\text{tfidf}(t_i, v_1), \text{tfidf}(t_i, v_2), \ldots, \text{tfidf}(t_i, v_V)]$. Then, the resulting TF-IDF matrix $(X \in R^{N \times V})$ serves as the high-dimensional representation of the comments.

Data Splitting and Handling Class Imbalance

The dataset is partitioned into training and testing subsets using an 80-20 split ratio $(X_{\text{train}}, y_{\text{train}}), (X_{\text{test}}, y_{\text{test}}) = \text{train\_test\_split}(X, y, \text{test\_size} = 0.2)$. To address class imbalance, the imbalance ratio (ρ) is computed as $\rho = \frac{\sum_{i=1}^{N} \mathbb{1}(y_i = 1)}{N}$, where $(\mathbb{1}(\cdot))$ is the indicator function. If significant imbalance is detected, Synthetic Minority Over-sampling Technique (SMOTE) is applied to generate synthetic samples of the minority class $x_{\text{new}} = x_{\text{minority}} + \lambda \cdot (x_{\text{neighbor}} - x_{\text{minority}})$, where (x_{minority}) is a minority class sample, (x_{neighbor}) is a randomly selected nearest neighbor, and $(\lambda \sim U(0,1))$.

Model Development

This research involves implementing and comparing several machine learning models. Let $(\mathcal{M})$ represent the set of models is presented in the equation 1.

$$\mathcal{M} = \{\text{LinearSVC}, \text{RandomForest}, \text{NaiveBayes}, \text{LogisticRegression}, \text{DecisionTree}, \text{KNN}, \text{GradientBoosting}, \text{XGBoost}, \text{LightGBM}, \text{VotingClassifier}\} \quad (1)$$

Each model $(M \in \mathcal{M})$ is trained to learn a function $(f_M : R^V \to \{0,1\})$. The LinearSVC model seeks to find a hyperplane that maximizes the margin between classes. The optimization problem for LinearSVC is defined as $\min_{w,b} \ \frac{1}{2}|w|^2 + C\sum_{i=1}^{N} \ \max(0, 1 - y_i(w \cdot x_i + b))$, where (w) is the weight vector, (b) is the bias term, and (C) is the regularization parameter. Naive Bayes assumes conditional independence between features, calculating the posterior probability for classification is $P(y \mid x) = \frac{P(y)\prod_{j=1}^{V} P(x_j \mid y)}{P(x)}$. In addition, RandomForest, an ensemble method, builds multiple decision trees on bootstrapped samples and aggregates their predictions $f_{\text{RF}}(x) = \arg\max_{c \in \{0,1\}} \sum_{t=1}^{T} \mathbb{1}(h_t(x) = c)$,
where (T) is the number of trees. GradientBoosting builds models sequentially, with each model correcting the residuals of its predecessor is $f^{(m)}(x) = f^{(m-1)}(x) + \eta h_m(x)$, where (η) is the learning rate. The VotingClassifier combines multiple models through majority voting $\hat{y} = \arg\max_{c \in \{0,1\}} \sum_{M \in \mathcal{S}} \mathbb{1}(f_M(x) = c)$.

Cross-Validation and Model Evaluation

To ensure robust evaluation, (k)-fold cross-validation is employed. The dataset is split into (k) folds, with each model $(M \in \mathcal{M})$ trained and validated across these folds. The performance metrics are calculated for each model as presented in the equation 2-5.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4)$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, **Number** 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4670

where (TP), (TN), (FP), and (FN) are derived from the confusion matrix. For evaluation, the model with the highest average F1-score across (k)-folds is selected as the best model. This model is then trained on the entire training set and evaluated on the test set X_{test} and y_{test}, The final evaluation includes metrics such as accuracy, precision, recall, F1-score, and the confusion matrix to provide a comprehensive performance assessment.

RESULT

This section presents the results obtained from the cross-validation of various machine learning models, followed by an evaluation of the best-performing model on the test set. The analysis includes a comparison of model performance based on accuracy, precision, recall, and F1-score. The discussion focuses on interpreting these metrics, assessing the strengths and limitations of each model, and highlighting the significance of the best-performing model for spam detection on YouTube comments.

To ensure a robust evaluation, a stratified 10-fold cross-validation was performed for each of the candidate models. The cross-validation results, summarized in the table below, reveal the performance metrics of each model, including accuracy, precision, recall, and F1-score as presented in the table 1. Among the evaluated models, Linear Support Vector Classifier (LinearSVC) achieved the highest performance across all metrics, with an accuracy of 95.33%, precision of 95.46%, recall of 95.33%, and an F1-score of 95.32%. This indicates that LinearSVC effectively distinguishes between spam and legitimate comments with a high degree of accuracy and balance. RandomForest closely followed LinearSVC, with an accuracy of 95.07% and similar values for precision, recall, and F1-score, demonstrating the robustness of ensemble learning methods in this classification task. DecisionTree, LightGBM, and VotingClassifier models also exhibited strong performance, with accuracy rates of 94.56% and F1-scores above 94.5%, suggesting that these models are capable of capturing complex patterns within the dataset. The VotingClassifier, which combines multiple models through majority voting, achieved a high precision of 94.75%, indicating its ability to leverage the strengths of its constituent models. However, its performance was marginally lower than that of LinearSVC and RandomForest, suggesting that the ensemble of base models did not outperform the best individual model in this context.

Table 1. Comparison of Performance Result

Model	Accuracy	Precision	Recall	F1 Score
LinearSVC	0.953265	0.954646	0.953265	0.953231
RandomForest	0.950704	0.952955	0.950704	0.950645
DecisionTree	0.945583	0.945795	0.945583	0.945577
LightGBM	0.945583	0.946588	0.945583	0.945554
VotingClassifier	0.945583	0.947495	0.945583	0.945527
XGBoost	0.939181	0.939709	0.939181	0.939164
LogisticRegression	0.937900	0.940251	0.937900	0.937820
GradientBoosting	0.935980	0.939723	0.935980	0.935847
KNN	0.911012	0.913887	0.911012	0.910862
NaiveBayes	0.763124	0.763171	0.763124	0.763116

XGBoost, LogisticRegression, and GradientBoosting displayed moderate performance, with accuracy scores ranging from 93.6% to 93.9%. These models, while slightly less accurate than the top performers, still offer competitive results and demonstrate the effectiveness of different learning paradigms in text classification. Notably, the NaiveBayes classifier had the lowest performance, with an accuracy of 76.31%, precision of 76.32%, recall of 76.31%, and an F1-score of 76.31%. This outcome is consistent with the limitations of the NaiveBayes model, which assumes feature independence, an assumption that may not hold in the context of complex textual data like YouTube comments. Given its superior cross-validation performance, LinearSVC was selected as the best model and subsequently evaluated on the test set to assess its generalization capabilities. The confusion matrix and performance metrics for LinearSVC on the test set are as presented in table 2.

Table 2. Confusion matrix and Performance of LinearSVC

Class	Precision	Recall	F1-score	Support
0	0.93	0.98	0.95	170
1	0.99	0.94	0.96	221
accuracy				391
macro avg	0.96	0.96	0.96	391
weighted avg	0.96	0.96	0.96	391

LinearSVC achieved an accuracy of 96% on the test set, confirming its ability to generalize well to unseen data. The precision for spam comments (class 1) was exceptionally high at 0.99, indicating that the model has a strong capability to correctly identify spam comments without generating many false positives. The recall for spam comments

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 4, Number 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4670

was 0.94, signifying that the model successfully identified 94% of all actual spam comments in the test set. This balance between precision and recall is reflected in the F1-score of 0.96, which provides a single metric that considers both false positives and false negatives. For legitimate comments (class 0), the precision was 0.93 and the recall was 0.98, leading to an F1-score of 0.95. These values indicate that while LinearSVC is slightly more conservative in classifying legitimate comments as non-spam, it effectively minimizes the false negative rate for legitimate comments. The overall performance suggests that LinearSVC is highly effective in handling the inherent trade-offs between precision and recall, making it a suitable choice for spam detection in the context of YouTube comments.

DISCUSSION

The high performance of LinearSVC can be attributed to its ability to find an optimal hyperplane in the high-dimensional feature space created by TF-IDF vectorization. By maximizing the margin between spam and legitimate comments, LinearSVC minimizes misclassifications, especially when dealing with complex and imbalanced data. The results show that LinearSVC not only excels in cross-validation but also maintains its performance on unseen data, highlighting its robustness and reliability for this classification task. RandomForest, with its ensemble learning approach, also performed exceptionally well, underscoring the importance of leveraging multiple decision boundaries to improve classification accuracy. However, its slight underperformance compared to LinearSVC suggests that while ensemble methods are powerful, they may not always capture the intricate linear separations that an SVM can provide in high-dimensional spaces.

The VotingClassifier, which aimed to combine the strengths of multiple models, yielded high precision but did not surpass LinearSVC. This outcome indicates that while ensemble strategies like majority voting can enhance performance, the selection and combination of base models are crucial. In this study, LinearSVC's straightforward yet effective approach provided a more decisive separation of classes than the combined efforts of the VotingClassifier. NaiveBayes' lower performance was expected, given its assumption of feature independence, which is unlikely to hold true in natural language processing tasks involving complex word dependencies and contextual meanings. The results reinforce the notion that more sophisticated models, capable of capturing nuanced patterns in the text, are necessary for accurate spam detection. Overall, the findings demonstrate that LinearSVC is a highly effective model for spam detection in YouTube comments, achieving a commendable balance between precision and recall. The high accuracy and F1-score indicate that this model can serve as a reliable component in automated moderation systems, reducing the prevalence of spam and improving the overall quality of user interactions on the platform. Further research could explore the integration of deep learning models to capture even more complex patterns in the data, potentially enhancing the detection of more subtle forms of spam.

CONCLUSION

This research aimed to develop an effective machine learning-based system for detecting spam comments on YouTube, a task that has become increasingly critical due to the growing prevalence of unsolicited and potentially harmful content on the platform. Through a systematic evaluation of various machine learning models, including traditional classifiers, ensemble methods, and a Voting Classifier, the study identified Linear Support Vector Classifier (LinearSVC) as the most effective model for this task. LinearSVC achieved the highest performance across multiple metrics, including an accuracy of 95.33% and an F1-score of 95.32% in cross-validation, and maintained an impressive accuracy of 96% on the test set. The superior performance of LinearSVC can be attributed to its ability to find an optimal hyperplane in the high-dimensional feature space created by TF-IDF vectorization, effectively maximizing the margin between spam and legitimate comments. This model demonstrated a remarkable balance between precision and recall, indicating its robustness in both identifying true spam comments and minimizing false positives. The consistent performance of LinearSVC across both the training and testing phases highlights its potential as a reliable tool for automated spam detection in large-scale comment moderation systems like YouTube.

RandomForest and other ensemble models, such as LightGBM and the VotingClassifier, also exhibited strong performance, underscoring the effectiveness of ensemble learning in capturing complex patterns within textual data. However, the marginal differences in performance between these models and LinearSVC suggest that, while ensemble strategies can enhance classification accuracy, simpler linear models may sometimes offer a more decisive separation of classes in high-dimensional text spaces. The NaiveBayes classifier, with its relatively lower performance, reaffirmed the limitations of assuming feature independence in natural language processing tasks, particularly in the context of nuanced and context-dependent spam content. The findings of this study have several implications. First, they demonstrate that linear models, when coupled with effective feature engineering techniques such as TF-IDF vectorization, can serve as a powerful approach for spam detection in online comment sections. Second, the research highlights the importance of selecting appropriate machine learning models based on the nature of the data and the specific requirements of the application domain. In the context of YouTube comments, where the volume and variety of content are vast, a model like LinearSVC offers a balance between computational efficiency and classification accuracy, making it suitable for real-time deployment.

E-ISSN : 2807-9035
Volume 4, **Number** 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4670

While the results are promising, this research also points to potential areas for further exploration. Future work could investigate the integration of deep learning models, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), which are known for their ability to capture intricate patterns in text data. Additionally, incorporating more advanced natural language processing techniques, such as word embeddings and contextual embeddings like BERT, could further enhance the detection of more subtle forms of spam. Moreover, the exploration of hybrid models that combine the strengths of traditional machine learning and deep learning approaches may yield even more robust solutions.

REFERENCES

Abbasi, A., Dobolyi, D., Vance, A. & Zahedi, F. M. (2021). The phishing funnel model: a design artifact to predict user susceptibility to phishing websites. Information Systems Research, 32(2), 410–436.

Abkenar, S. B., Kashani, M. H., Akbari, M. & Mahdipour, E. (2023). Learning textual features for Twitter spam detection: A systematic literature review. Expert Systems with Applications, 228, 120366.

Agarwal, R., Dhoot, A., Kant, S., Bisht, V. S., Malik, H., Ansari, M. F., Afthanorhan, A. & Hossaini, M. A. (2024). A novel approach for spam detection using natural language processing with AMALS models. IEEE Access.

Ahmed, S. F., Alam, M. S. Bin, Hassan, M., Rozbu, M. R., Ishtiak, T., Rafa, N., Mofijur, M., Shawkat Ali, A. B. M. & Gandomi, A. H. (2023). Deep learning modelling techniques: current progress, applications, advantages, and challenges. Artificial Intelligence Review, 56(11), 13521–13617.

Akinyelu, A. A. (2021). Advances in spam detection for email spam, web spam, social network spam, and review spam: ML-based and nature-inspired-based techniques. Journal of Computer Security, 29(5), 473–529.

Al Zoubi, A. M. & others. (2024). Spam Reviews Detection Models in Multilingual Contexts applying Sentiment Analysis, Metaheuristics, and Advanced Word Embedding.

Alkhamees, M., Alsaleem, S., Al-Qurishi, M., Al-Rubaian, M. & Hussain, A. (2021). User trustworthiness in online social networks: A systematic review. Applied Soft Computing, 103, 107159.

Andresini, G., Iovine, A., Gasbarro, R., Lomolino, M., de Gemmis, M. & Appice, A. (2022). Review Spam Detection using Multi-View Deep Learning Combining Content and Behavioral Features. ItaDATA, 87–98.

Appel, G., Grewal, L., Hadi, R. & Stephen, A. T. (2020). The future of social media in marketing. Journal of the Academy of Marketing Science, 48(1), 79–95.

Barushka, A. (2020). Machine Learning Techniques in Spam Filtering.

Bazzaz Abkenar, S., Mahdipour, E., Jameii, S. M. & Haghi Kashani, M. (2021). A hybrid classification method for Twitter spam detection based on differential evolution and random forest. Concurrency and Computation: Practice and Experience, 33(21), e6381.

Evans, D., Bratton, S. & McKee, J. (2021). Social media marketing. AG Printing \& Publishing.

Fafalios, S., Charonyktakis, P. & Tsamardinos, I. (2020). Gradient boosting trees. Gnosis Data Analysis PC, 1.

Gaafar, A. S., Dahr, J. M. & Hamoud, A. K. (2022). Comparative analysis of performance of deep learning classification approach based on LSTM-RNN for textual and image datasets. Informatica, 46(5).

Galli, F., Loreggia, A. & Sartor, G. (2022). The Regulation of Content Moderation. International Conference on the Legal Challenges of the Fourth Industrial Revolution, 63–87.

Gasparetto, A., Marcuzzo, M., Zangari, A. & Albarelli, A. (2022). A survey on text classification algorithms: From text to predictions. Information, 13(2), 83.

Gongane, V. U., Munot, M. V & Anuse, A. D. (2022). Detection and moderation of detrimental content on social media platforms: current status and future directions. Social Network Analysis and Mining, 12(1), 129.

Gunturi, S. K. & Sarkar, D. (2021). Ensemble machine learning models for the detection of energy theft. Electric Power Systems Research, 192, 106904.

Hakak, S., Alazab, M., Khan, S., Gadekallu, T. R., Maddikunta, P. K. R. & Khan, W. Z. (2021). An ensemble machine learning approach through effective feature extraction to classify fake news. Future Generation Computer Systems, 117, 47–58.

Hassani, H., Beneki, C., Unger, S., Mazinani, M. T. & Yeganegi, M. R. (2020). Text mining in big data analytics. Big Data and Cognitive Computing, 4(1), 1.

Hussain, K., Khan, M. L. & Malik, A. (2024). Exploring audience engagement with ChatGPT-related content on YouTube: Implications for content creators and AI tool developers. Digital Business, 4(1), 100071.

Jahan, M. S. & Oussalah, M. (2023). A systematic review of hate speech automatic detection using natural language processing. Neurocomputing, 546, 126232.

Jain, A. K., Sahoo, S. R. & Kaubiyal, J. (2021). Online social networks security and privacy: comprehensive review and analysis. Complex \& Intelligent Systems, 7(5), 2157–2177.

Jáñez-Martino, F., Alaiz-Rodr\'\iguez, R., González-Castro, V., Fidalgo, E. & Alegre, E. (2023). A review of spam email detection: analysis of spammer strategies and the dataset shift problem. Artificial Intelligence Review, 56(2), 1145–1173.

E-ISSN : 2807-9035
Volume 4, **Number** 2, November 2024
https://doi.org/10.47709/brilliance.v4i2.4670

Kavzoglu, T. & Teke, A. (2022). Advanced hyperparameter optimization for improved spatial prediction of shallow landslides using extreme gradient boosting (XGBoost). Bulletin of Engineering Geology and the Environment, 81(5), 201.

Khan, H., Asghar, M. U., Asghar, M. Z., Srivastava, G., Maddikunta, P. K. R. & Gadekallu, T. R. (2021). Fake review classification using supervised machine learning. Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10--15, 2021, Proceedings, Part IV, 269–288.

Liu, J., Singhal, T., Blessing, L. T. M., Wood, K. L. & Lim, K. H. (2021). Crisisbert: a robust transformer for crisis classification and contextual crisis embedding. Proceedings of the 32nd ACM Conference on Hypertext and Social Media, 133–141.

Manca, S., Bocconi, S. & Gleason, B. (2021). "Think globally, act locally": A glocal approach to the development of social media literacy. Computers \& Education, 160, 104025.

Mohammed, A. & Kora, R. (2023). A comprehensive review on ensemble deep learning: Opportunities and challenges. Journal of King Saud University-Computer and Information Sciences, 35(2), 757–774.

Nascimento, F. R. S., Cavalcanti, G. D. C. & Da Costa-Abreu, M. (2023). Exploring automatic hate speech detection on social media: a focus on content-based analysis. SAGE Open, 13(2), 21582440231181310.

Rao, S., Verma, A. K. & Bhatia, T. (2021). A review on social spam detection: Challenges, open issues, and future directions. Expert Systems with Applications, 186, 115742.

Rao, S., Verma, A. K. & Bhatia, T. (2023). Hybrid ensemble framework with self-attention mechanism for social spam detection on imbalanced data. Expert Systems with Applications, 217, 119594.

Rastogi, A., Mehrotra, M. & Ali, S. S. (2020). Effective opinion spam detection: A study on review metadata versus content. Journal of Data and Information Science, 5(2), 76–110.

Reshma, P. K. (2020). Soft computing approaches to domain specific information retrieval in the semantic web. University of Calicut.

Salman, M., Ikram, M. & Kaafar, M. A. (2024). Investigating Evasive Techniques in SMS Spam Filtering: A Comparative Analysis of Machine Learning Models. IEEE Access.

Saumya, S. & Singh, J. P. (2022). Spam review detection using LSTM autoencoder: an unsupervised approach. Electronic Commerce Research, 22(1), 113–133.

Teng, S., Khong, K. W., Pahlevan Sharif, S. & Ahmed, A. (2020). YouTube Video comments on healthy eating: descriptive and predictive analysis. JMIR Public Health and Surveillance, 6(4), e19618.

Wang, J., Xue, D. & Shi, K. (2021). An ensemble framework for spam detection on social media platforms. International Journal of Machine Learning and Computing, 11(1), 77–84.

Wang, M., Fu, W., He, X., Hao, S. & Wu, X. (2020). A survey on large-scale machine learning. IEEE Transactions on Knowledge and Data Engineering, 34(6), 2574–2594.

Yang, X., Yang, K., Cui, T., Chen, M. & He, L. (2022). A study of text vectorization method combining topic model and transfer learning. Processes, 10(2), 350.

Zeakis, A., Papadakis, G., Skoutas, D. & Koubarakis, M. (2023). Pre-trained embeddings for entity resolution: an experimental analysis. Proceedings of the VLDB Endowment, 16(9), 2225–2238.

Zhang, M. (2024). Ensemble-Based Text Classification for Spam Detection. Informatica, 48(6).