YOUTUBE COMMENT SPAM DETECTION

Anusha Nandy
Department of Computer Science
University of Alabama at Birmingham
Alabama, AL 35233, USA
nandya@uab.edu

Siddharth Bhandary
Department of Computer Science
University of Alabama at Birmingham
Alabama, AL 35233, USA
bhandary@uab.edu

Pavan Agnihotri
Department of Computer Science
University of Alabama at Birmingham
Alabama, AL 35233, USA
pagnihot@uab.edu

Nikhil Kanamadi
Department of Computer Science
University of Alabama at Birmingham
Alabama, AL 35233, USA
kanamadi@uab.edu

Please click on the underlined link to our presentation video shared on OneDrive: **Please click here**

ABSTRACT

YouTube, since its inception in 2005, has played a pivotal role in the distribution of knowledge and content to a global audience. However, with the exponential growth of its user base, the platform has become a target for malicious actors seeking to post spam and harmful comments, which can hamper a users experience and tarnish the reputation of content creators. In this paper, we propose novel approaches to differentiate between legitimate and spam comments on YouTube. We explore the application of various machine learning models, including Logistic Regression, Support Vector Machines (SVM), Naive Bayes, Decision Trees, Random Forests, Convolutional Neural Networks (CNNs), and BERT, which have demonstrated success in detecting malicious comments in other domains. Our work aims to develop more robust and adaptive models that can better address the evolving nature of spam and improve the accuracy of comment moderation on YouTube.

1 INTRODUCTION

YouTube has evolved into one of the most prominent platforms for content consumption and knowledge dissemination, with more than 2 billion active users worldwide. As the second-largest search engine globally, it offers videos on virtually every topic imaginable, ranging from educational content to entertainment. The platform has also revolutionized the way people interact with video media, allowing users to upload, comment, and share content with unprecedented ease. With more than 3.5 million videos uploaded daily and hundreds of millions of comments posted, YouTube's vast and diverse user base contributes to a dynamic and active ecosystem. However, the platform's open-commenting system, while enabling rich interaction, also poses significant challenges.

The freedom to comment on videos has created a double-edged sword: while it fosters open discussion and engagement, it also opens the door for spam and malicious activity. Spam comments, defined as irrelevant or unsolicited messages often intended to promote malicious links, advertisements, or misleading content, can detract from the user experience and tarnish the integrity of the platform. "Ham," in contrast, refers to legitimate, meaningful comments that contribute to the conversation. Although YouTube employs automated systems to detect spam, spammers continually refine their techniques to circumvent these filters, posing a constant challenge to maintaining the quality of content. In addition to textual spam, bots have also been deployed to artificially inflate view counts and engage in fraudulent monetization schemes, further complicating the task of distinguishing genuine user interaction from harmful activity.

The presence of spam can be particularly detrimental to less experienced users, such as elderly individuals, who may struggle to differentiate between legitimate and spam comments. This demographic, often less familiar with on-line platforms, may unknowingly click on malicious links or fall prey to deceptive tactics that compromise their security or diminish their overall experience on the platform. Furthermore, the prevalence of spam can erode trust in YouTube's comment system, discourage constructive discussion, and potentially drive users away.

Given the growing scale and sophistication of these challenges, this paper aims to explore new methodologies to effectively filter spam comments, thereby safeguarding the integrity of YouTube's platform. By leveraging advanced machine learning techniques, we propose models capable of more accurately distinguishing between ham and spam comments. This research seeks to enhance YouTube's ability to moderate comments and provide a safer, more trustworthy environment for its diverse user base.

2 Related Works

2.1 Oh, H. (2021)

A cascaded ensemble machine learning model was created for YouTube spam comment detection, emphasizing how successful it is in comparison to more conventional techniques. The study uses a variety of classifiers to gather and categorize comments from well-known films and presents the ESM-S model, which combines predictions to increase accuracy. The findings show that this model works better than others in spam identification, indicating the possibility of further improvement using cutting-edge methods in further studies.(9)

2.2 Aiyar, S., & Shetty, N. P. (2018)

It tackles how to spot spam comments on YouTube using n-gram approaches. Because YouTube has so much user-generated video, spam comments—which are frequently submitted by automated bots—have grown to be a serious problem. In order to improve spam detection accuracy and get a high F1 score, the authors suggest an approach that makes use of n-grams and machine learning algorithms. They go over the difficulties in choosing suitable n values as well as the consequences of employing a Bag of Words strategy in the absence of traditional preprocessing methods. The report emphasizes how crucial efficient spam filtering is to enhancing the platform's user experience.(10)

2.3 Xiao, A. S., & Liang, Q. (2024)

It examines the performance of eight machine learning techniques for identifying spam in YouTube video comments. The models that were assessed include the voting classifier, Random Forest, Naive Bayes, Logistic Regression, KNN, MLP, SVC, and Decision Tree. The models are evaluated using performance measures including recall, F1 score, accuracy, precision, and AUC; Random Forest and SVC produce the best overall results. Through efficient detection and removal of spam comments, the study highlights the significance of these methods in enhancing content moderation and user experience on websites such as YouTube.(11)

2.4 Valpadasu, H. et. al (2023)

It focuses on locating and eliminating YouTube spam comments, which may contain commercial or irrelevant information. It uses a plethora of machine learning classifiers, such as decision trees, logistic regression, and neural networks, to efficiently detect spam and classifies user comments according to their applicability to the video content. The study identifies the shortcomings of current spam detection systems and suggests a Naive Bayes classifier-based, more accurate approach. In comparison to earlier tactics, the results show an 18 % increase in spam detection accuracy, with the goal of improving platform user experience by lowering unsolicited spam comments.(12)

2.5 SAMSUDIN, N. AL (2019)

It discusses the use of Logistic Regression (LR) and Support Vector Machine (SVM) for spam detection in YouTube comments. Logistic Regression, a popular and straightforward classification technique, predicts discrete probabilities, such as distinguishing between spam and non-spam comments. It excels due to its simplicity and strong performance across varied datasets. In experiments, Logistic Regression achieved an accuracy of 85.42% in Weka and 80.88% in RapidMiner, proving its effectiveness. Support Vector Machine (SVM) was also used in the study as a robust supervised learning classifier. It constructs hyperplanes in high-dimensional spaces to differentiate classes effectively. While SVM showed promising accuracy, its results were slightly lower than Logistic Regression in Weka (85.04%) and significantly lower in RapidMiner (73.68%). Both techniques, however, demonstrated the importance of feature selection and preprocessing for enhancing spam detection accuracy.(16)

2.6 ABD, T. ET. AL (2018)

It looks into YouTube's comment spam problem and shows how inadequate the platform's integrated filtering mechanism is. via a dataset taken from YouTube via its Data API, it contrasts several spam filtering strategies, such as genetic algorithms, machine learning, and statistical approaches. The study found that certain algorithms, such as AGA, ICA, and k-NN, were especially successful, achieving above 90% accuracy with the majority of algorithms. Potential enhancements to YouTube's spam filtering and the creation of browser extensions to assist comment moderation are suggested by the research.(13)

2.7 ALSALEH, M. ET. AL (2015)

This paper describes the creation of a machine learning-based comment spam detection technology that distinguishes between spam and benign comments is covered in the study. It also goes into detail the architecture of the system, which comprises elements for text preprocessing, feature extraction, XML dataset parsing, and the use of several machine learning classifiers such as neural networks, decision trees, support vector machines, and random forests.The results from the paper showed that neural networks (MLP-GD) and random forests perform best among the classifiers tested. The authors also provide a newly labeled dataset for future research and highlight the need for balanced datasets to improve detection performance.(15)

2.8 CHOWDURY, R. ET. AL (2013)

This paper addresses the growing issue of video spammers on the site. It highlights the drawbacks of YouTube's current spam detection techniques, which mostly target large numbers of messages and comments. To distinguish between real users and spammers, the authors propose a comprehensive method that makes use of a variety of user and video characteristics. The usefulness of three data mining methods such as decision tree, clustering, and naïve bayes in detecting spam is evaluated in this study. It finds that the Naïve Bayes and Decision Tree models perform better on bigger test datasets. The goal of the project is to enhance YouTube's user experience by better detecting undesired content.(14)

3 METHODS

We implemented and compared multiple machine learning approaches, ranging from traditional algorithms to deep learning models. The implementation process involved several key steps:

First, we preprocessed the YouTube comments by removing URLs, mentions (@username), special characters, and converting text to lowercase.

The data was split into training (80%) and testing (20%) sets to evaluate model performance.

TF-IDF was then employed to convert raw YouTube comments into numerical features, allowing machine learning models to understand and classify them effectively. Each YouTube comment was transformed into a sparse vector, where each feature represented the importance of a specific word

in the context of the entire dataset. This vectorization step was crucial for the models to learn meaningful patterns from the text data.

We chose TF-IDF over simpler techniques like bag-of-words because TF-IDF helps reduce the impact of commonly occurring words (e.g., ”the,” ”and,” ”is”) that do not contribute much to the meaning of the text, and instead emphasizes more unique and informative words that are more likely to indicate spam. The number of features (terms) in the TF-IDF matrix was limited to 5000 to control dimensionality and computational efficiency. The resulting TF-IDF vectors were sparse, with each feature representing the importance of a word in a given comment, based on its frequency in the comment and the inverse frequency across the entire dataset. This allowed the models to focus on significant terms, such as ”buy,” ”link,” or ”discount,” which often indicate spammy behavior in YouTube comments.

After applying TF-IDF to the YouTube comments, the resulting feature vectors were fed into various machine learning models as listed below:

3.1 Logistic Regression Model

Logistic Regression is a linear model designed for binary classification tasks. It works by estimating the probability that a given input belongs to a specific class using the logistic function, also known as the sigmoid function. This model calculates the log-odds of a sample belonging to the spam class (1) and applies a threshold to make predictions. The model operates on the principle of linear decision boundaries, where it computes the weighted sum of input features and applies a threshold (commonly 0.5) to assign class labels (Scikit-learn developers - Logistic Regression). (3)

We implemented logistic regression as our baseline model, using default parameters except for max_iterations set to 1000. This model served as a simple yet effective starting point for comparing more complex approaches.

3.2 Support Vector Machine (SVM)

It is a supervised learning model which is able to perform linear and non-linear classification tasks. The main focus for an SVM is its ability is to obtain an optimal hyperplane which is able to maximize the margin between two classes. The margin is the distance between the hyperplane and the support vectors which are the closest data points from each class. SVM can employ various kernel functions to transform the data and handle complex decision boundaries:

- Linear Kernel: Finds a straight hyperplane for linearly separable data.
- Polynomial Kernel: Captures polynomial relationships by mapping data into a higher-dimensional space.
- RBF (Radial Basis Function) Kernel: Data is mapped into an infinite-dimensional space, effectively handling non-linear patterns through the creation of complex decision boundaries.

The performance of the model depends on the type of kernel used in an SVM. The linear kernel is suitable for simpler, linearly separable datasets, while the RBF kernel excels in capturing intricate, non-linear relationships. We experimented with all the three kernel functions - linear, polynomial, and RBF - to determine the most effective approach.(Scikit-learn developers - SVM). (4)

3.3 Naive Bayes Theorem (NB)

It is a probabilistic machine learning algorithm based on the Bayes' Theorem. It is popular for classification tasks in areas such as spam detection, medical diagnosis, and sentiment analysis. Despite its simplicity, Naive Bayes often performs well even with small datasets and can handle high-dimensional data efficiently.

We employed the Multinomial Naive Bayes classifier with a smoothing parameter ($\alpha = 0.5$), which was particularly chosen for its efficiency with text classification tasks and our dataset's characteristics (Zhang,2004).(5)

3.4 Decision Tree

It is a supervised machine learning algorithm which is used for classification tasks, such as predicting whether an Email or YouTube comment is Spam or Not Spam. (Devroye,1996).(2)

It models decisions based on a series of questions about the features of the data and predicts the class label.

Our implementation used entropy as the splitting criterion with a maximum depth of 15, chosen to balance between model complexity and overfitting on our YouTube comment dataset.

3.5 Random Forest

The Random Forest classifier can control overfitting and lead to an improved prediction accuracy due to its ability to build multiple decision trees and combine their outputs (Baladram, 2024).(7)

It was used for this project as it can handle sparse and high dimensional data with the aid of TF-IDF vectorization which was also able to find out how important a word is compared to the entire collection of words in the comments.

Random forest was also used for this project due to its robustness to noisy or imbalanced data. YouTube spam comments can generally contain a bunch of random letters, emojis or words that are either related to the content of the video or to promote something that is completely unrelated to the video that it had commented under.

We configured the Random Forest classifier with 100 estimators and a maximum depth of 15, specifically tuned to handle the high-dimensional nature of our TF-IDF vectorized comment data.

The Random Forest classifier is able to learn a variety of decision boundaries enabling it to capture nuanced differences between spam and non-spam comments with the help of feature bagging and random splits.

3.6 XGBoost

XGBoost is a powerful machine learning algorithm based on gradient boosting, a technique that builds an ensemble of decision trees sequentially, where each tree corrects the errors made by the previous ones. It is particularly known for its efficiency, scalability, and strong predictive performance, especially when handling large and complex datasets. XGBoost works by iteratively adding new trees to minimize errors, focusing on the mistakes (residuals) made by earlier trees. Unlike traditional gradient boosting, XGBoost incorporates an optimized tree construction algorithm that uses regularization to prevent overfitting and efficiently handles missing values (Shabadi et. al, 2023).(17)

In this project, the XGBoost model was initialized with default parameters, including label encoding for the target classes. The model was then trained on TF-IDF transformed text data, which is often used in text classification tasks. XGBoost is especially well-suited for high-dimensional datasets, such as spam detection, because of its ability to handle non-linear relationships, reduce overfitting through regularization, and work efficiently with sparse data. In our experiments, XGBoost demonstrated strong performance, performing competitively with other models and providing a solid approach for classifying YouTube comments as spam or not spam.

3.7 BERT

For this project, BERT was fine-tuned to classify YouTube comments as spam or non-spam, leveraging its deep contextual understanding of language. The pre-trained BERT model was used to process the raw comments, and we fine-tuned it on the training set of TF-IDF-encoded comments. Given BERT's ability to understand the bidirectional context of words, it was particularly effective at capturing nuanced patterns in comments that contain complex spam markers, such as links, emojis, and repetitive phrases. The model was trained using the same training data and then used to predict the class labels for the test set. BERT significantly improved classification accuracy, especially for comments with complicated structures or subtle language cues that other models struggled to interpret.

However, the computational cost was higher than traditional machine learning models, making it suitable for tasks where accuracy is prioritized over efficiency.

3.8 Convolutional Neural Network (CNN)

Here CNN was used as it can learn hierarchical patterns in text data directly from the input. It doesn't depend on manually engineered features as seen in other machine learning models as it is able to extract contextual patterns, semantic relationships and n-grams. This is features are important for identifying YouTube comments which have hyperlinks, emojis, marketing keywords and repetitive phrases.

The CNN architecture in this project has embedding layers for representing tokens and multiple convolutional filters with varying kernel sizes to be able to capture patterns of different lengths.

Also it is to be noted that max-pooling layers used help highlight the most significant features within each comment. This makes the model to focus on relevant text segments in the comments. Dropout and fully connected layers further enhance the model's generalization capabilities for the binary classification task (IBM, n.d). (6)

By training the CNN end-to-end on tokenized YouTube comments, which are then preprocessed using a pretrained BERT tokenizer, the model is then able to learn non-linear patterns in the data. This helps to capture both local and global features within comments.

4 Dataset

The dataset used is the Youtube Comments Spam Dataset which was obtained from Kaggle. It has 1956 unique comments consisting of different languages (English and Korean), links and emojis. . This dataset has 6 features namely COMMENT_ID, AUTHOR, DATE, CONTENT, VIDEO_NAME and CLASS. The feature COMMENT_ID uniquely identifies each comment in the dataset, AUTHOR is the username of the person who posted a comment and DATE is when they posted the comment. The feature VIDEO_NAME as it suggests is the name of the video that the comments are under. The videos that were used are 'PSY - GANGNAM STYLE', 'Katy Perry - Dark Horse', "LMFAO - Party Rock Anthem", "Eminem feat. Rihanna - Love the Way you Lie", and "Shakira - Waka Waka". CLASS is a binary label where '1' represents that it is a spam comment and '0' represents that it is a non-spam comment (Waheed, 2024).(1)

Video Name	**YouTube ID**	**Spam**	**Non-Spam**	**Total**
PSY - GANGNAM STYLE	9bZkp7q19f0	175	175	350
Katy Perry - Roar	CevxZvSJLk8	175	175	350
LMFAO - Party Rock Anthem	KQ6zr6kCPj8	236	202	438
Eminem - Love The Way You Lie	ue1Hw8o7U	245	203	448
Shakira - Waka Waka	pRpeEdMmmQ0	174	196	370
Total		**1005**	**978**	**1956**

Table 1: Distribution of Spam and Non-spam comments across different youtube videos

The reason why this dataset was used for classifying spam comments is because of its diverse linguistic patterns that were used. Each spam comment had traits such as repetitive phrases, links, and marketing messages along with the dataset containing non-spam comments helped the model to learn the difference between the two. The size of the dataset also allowed the model to have enough examples to understand how to distinguish between the spam and non-spam comments and also made the model be able to use slightly less computational resources.

5 Results and Evaluation

In this study, we conducted a comprehensive evaluation of several machine learning models for YouTube spam comment detection. The experiments were executed in a Google Colab environment using Python 3.8, with the dataset split into 80% training and 20% test sets to ensure robust evaluation.

5.1 Performance Metrics

Table 2 presents the comprehensive performance metrics for each algorithm.

Algorithm	**Accuracy**
Logistic Regression	0.9413
SVM (Linear)	0.9489
Random Forest	0.9235
Naive Bayes	0.9133
SVM (Polynomial)	0.9030
SVM (RBF)	0.9387
Decision Tree	0.9362
CNN	0.9719
XGBoost	0.9413
BERT	0.9745

Table 2: Performance Metrics for Each Algorithm

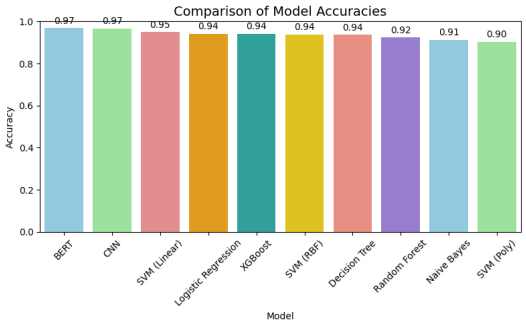

Figure 1: Comparison of Model Accuracies

5.2 Model Analysis

BERT achieved the highest accuracy at 97.45%, followed closely by CNN at 97.19%. The superior performance of these deep learning approaches can be attributed to their ability to capture semantics, automatically learn hierarchical features, and its ability to effectively to handle context-dependent patterns.

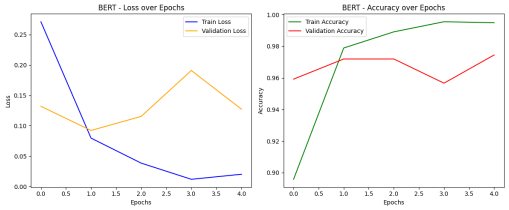

Figure 2: BERT - Loss and Accuracy over Epochs

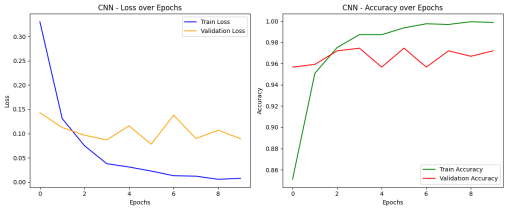

Figure 3: CNN - Loss and Accuracy over Epochs

Traditional machine learning models also performed well, with Linear SVM achieving 94.89% accuracy, demonstrating the effectiveness of simpler approaches for this task.

5.3 Hardware Specifications

All experiments were conducted using Google Colab with the following specifications:

- GPU: NVIDIA Tesla T4
- RAM: 12GB
- CPU: Intel Xeon @ 2.20GHz

6 CONCLUSION

The purpose of this project was to highlight the importance of classifying YouTube spam comments as they are currently ubiquitous in nature. Finding a way to classify between spam and non spam comments allows helps to maintain user security and to improve user experience. In this project, it is observed that by comparing the different machine learning models, we were able to obtain the highest accuracy with BERT with 97.45% accuracy and CNN with 97.19 %. Some of the limitations in YouTube spam classification model such as its dependence on labeled data for training and difficulties in understanding context-specific material can be solved by either using unsupervised learning to try to reduce its dependence on labeled data and the use of more advanced models such as RoBERTa to be able to understand diverse languages and contexts.

REFERENCES

[1] Waheed, A. (2024). Youtube Comments Spam Dataset, Version 1. Retrieved October 3, 2024, from https://www.kaggle.com/datasets/ahsenwaheed/youtube-comments-spam-dataset/data

[2] Devroye, L., Gyorfi, L., & Lugosi, G. (1996). A probabilistic theory of pattern recognition. Springer. Retrieved from https://www.szit.bme.hu/~gyorfi/pbook.pdf

[3] Scikit-learn developers. (n.d.). *LogisticRegression*. Scikit-learn. Retrieved from https://scikit-learn.org/1.5/modules/generated/sklearn.linear_model.LogisticRegression.html

[4] Scikit-learn developers. (n.d.). *Support vector machines (SVM)*. Scikit-learn. Retrieved from https://scikit-learn.org/1.5/modules/svm.html

[5] Zhang, H. (2004). The optimality of naive Bayes. Retrieved from https://aaai.org/papers/flairs-2004-097/

[6] IBM. (n.d.). *Convolutional neural networks*. Retrieved from https://www.ibm.com/topics/convolutional-neural-networks

[7] Baladram, S. (2024, November 7). Random forest explained: A visual guide with code examples. Towards Data Science - Medium. Retrieved from https://towardsdatascience.com/random-forest-explained-a-visual-guide-with-code-examples-9f736a6e1b3c

[8] Pott, C. (2022, January 6). RoBERTa — Stanford CS224U Natural Language Understanding — Spring 2021. Stanford Online. Retrieved from https://www.youtube.com/watch?v=EZMOBbu_5b8

[9] Oh, H. (2021). A YouTube spam comments detection scheme using cascaded ensemble machine learning model. *IEEE Access*, *9*, 144121–144128. https://doi.org/10.1109/ACCESS.2021.3121508

[10] Aiyar, S., & Shetty, N. P. (2018). N-gram assisted YouTube spam comment detection. *Procedia Computer Science*, *132*, 174–182. https://doi.org/10.1016/j.procs.2018.05.181

[11] Xiao, A. S., & Liang, Q. (2024). Spam detection for YouTube video comments using machine learning approaches. *Machine Learning with Applications*, *16*, 100550. https://doi.org/10.1016/j.mlwa.2024.100550

[12] Valpadasu, H., Chakri, P., Harshitha, P., & Tarun, P. (2023). Machine learning-based spam comments detection on YouTube. In *2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 1234–1239). IEEE. https://doi.org/10.1109/ICICCS56967.2023.10142608

[13] Abd, T., Altabrawee, H., & Qaisar, S. (2018). YouTube spam comments detection using artificial neural network. *Journal of Engineering and Applied Sciences*, *13*(6), 9638–9642. https://doi.org/10.3923/jeasci.2018.9638.9642

[14] Chowdury, R., Adnan, M. N. M., Mahmud, G. A. N., & Rahman, R. M. (2013). A data mining-based spam detection system for YouTube. In *Eighth International Conference on Digital Information Management (ICDIM 2013)* (pp. 373–378). IEEE. https://doi.org/10.1109/ICDIM.2013.6694038

[15] Alsaleh, M., Alarifi, A., Al-Quayed, F., & Al-Salman, A. (2015). Combating comment spam with machine learning approaches. In *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)* (pp. 295–300). IEEE. https://doi.org/10.1109/ICMLA.2015.192

[16] Samsudin, N., Mohd Foozy, C. F., Alias, N., Shamala, P., Othman, N., & Wan Din, W. I. S. (2019). YouTube spam detection framework using naïve Bayes and logistic regression. *Indonesian Journal of Electrical Engineering and Computer Science*, *14*(3), 1508–1517. https://doi.org/10.11591/ijeecs.v14.i3.pp1508-1517

[17] Shabadi, L., Y. L., C., P., S., L., V. K., & Kashyap, U. (2023). YouTube spam detection scheme using stacked ensemble machine learning model. *2023 International Conference on Network, Multimedia and Information Technology (NMITCON)*, 1–7. https://doi.org/10.1109/NMITCON58196.2023.10276002