Vol. 11 No.2 Tahun 2025, pp 132-142
Jurnal Teknologi dan Manajemen Informatika
http:// http://jurnal.unmer.ac.id/index.php/jtmi
P-ISSN: 1693-6604 E-ISSN: 2580-8044

Comparative Performance of Machine Learning Algorithms for Detecting Online Gambling Promotional Comments on Youtube

Michael Angelo [1], Robet [2], Jackri Hendrik [3]

STMIK Time, Indonesia

Article Info

Article History
Received: 13-10-2025
Revised : 22-11-2025
Accepted: 26-11-2025

Keywords
Machine Learning;
Comment Detection;
Pseudo-Labeling;
Online Gambling Promotion;

✉ Corresponding Author
Michael Angelo,
STMIK Time
michaelangzz8998@gmail.com

ABSTRACT

Online-gambling promoters increasingly exploit YouTube comment sections, using text obfuscation, Unicode characters, emojis, irregular spacing, and symbols to evade automated moderation. This study aims to identify the most effective machine-learning algorithm for detecting such promotional comments by comparing models on standard metrics (precision, recall, F1-score, accuracy). We employ semi-supervised pseudo-labelling to expand the labelled set from 1,648 to 9,111 comments without additional manual annotation, admitting only high-confidence predictions. The pipeline includes customised character normalization, selective cleaning, tokenization, stopword removal, and Nazief–Adriani stemming, followed by TF–IDF feature extraction. Four algorithms are evaluated: Multinomial Naive Bayes, Logistic Regression, Random Forest, and Support Vector Machine, with hyperparameter optimization and class balancing via SMOTE. On a 1,823-sample test set, all models achieve over 98% accuracy; SVM yields the most balanced performance, resulting in the highest F1-score for the promotion class (0.9908). Confusion matrices and learning curves indicate stable behavior without overfitting or underfitting. We therefore recommend SVM for operational deployment in automated moderation of gambling-promotion comments on YouTube. These findings provide practical guidance for platform safety teams and suggest methodological baselines for similar NLP moderation tasks. Future work should explore ensemble and deep learning approaches, incorporate character and subword-level features, and further evaluate robustness under adversarial obfuscation and domain shift.

INTRODUCTION

YouTube has evolved beyond a medium for entertainment and information into an open, interactive space that facilitates user-to-user communication through its comments feature. However, since early 2025 there has been a significant rise in the misuse of the comments feature to massively disseminate promotions for online gambling sites, particularly on YouTube channels with high engagement levels, such as those of Deddy Corbuzier and Jess No Limit. These promotions are often obfuscated using various techniques, including the use of non-standard Unicode characters and character splitting to evade keyword-based filters [1], [2]. In addition, emojis are frequently employed as a covert language to convey harmful meanings that are not explicit in text [3].

This phenomenon generates several serious impacts. First, gambling-related comments may reach young audiences or other vulnerable groups, who represent a substantial portion of

This is an open access article under the CC BY-SA license.
DOI: https://doi.org/10.26905/jtmi.v11i2.16286

active YouTube users [4]. Promoters often exploit the curiosity of younger viewers by packaging promotional messages to resemble entertainment content, giveaways, or online games, making them appear harmless at first glance. Second, the spread of illegal content undermines platform integrity and damages the reputation of content creators whose videos are misused as promotional media [5]. Recent studies indicate that spam comments on YouTube directly disrupt user experience, reduce the quality of communication within the comment section, and negatively affect the perceived credibility of both channels and the platform as a whole [6]. Third, these promotional comments have the potential to influence the performance of YouTube's recommendation algorithm [7].

To evade detection, promoters typically employ various obfuscation techniques, such as homoglyph attacks (mixing Latin and non-Latin characters), leetspeak (e.g., replacing letters with numbers such as "sl0t"), zero-width spaces, and the insertion of unusual symbols or characters that interfere with standard tokenization. The inability of automated moderation systems to recognize such covert comments not only disrupts user comfort due to irrelevant messages but may also produce false engagement metrics, ultimately reducing user trust and overall interaction with the platform. A viable solution to these challenges is the development of automated, machine-learning-based classification approaches capable of distinguishing gambling-promotion comments from ordinary comments, including those that are visually or semantically modified [8], [9]. Furthermore, YouTube comments have become a rich data source for various NLP tasks such as sentiment analysis [10], [11], making accurate text processing increasingly essential. For example, Salsabila and Trianasari applied sentiment analysis and topic modeling on 4,961 Indonesian cosmetic product reviews to extract prevalent user opinions [12]. With such automation, the burden of inefficient manual moderation can be reduced, while providing more effective protection against the spread of illegal promotional content.

Prior research on classifying comments or text using machine learning is extensive, with multiple literature reviews confirming the effectiveness of these approaches [13], [14]. For example, one study applied the Naive Bayes algorithm to detect spam comments on online forums, although it did not address the complexity of Unicode characters and symbols commonly used by spammers [15]. Nevertheless, Naïve Bayes has demonstrated high accuracy in classifying Indonesian text; for instance, one study reported 86.67% accuracy on an Indonesian Twitter sentiment analysis task [16]. Other studies have implemented SVM and Random Forest, including variants such as Extra Trees [17], to classify hate speech and spam on social media, reporting strong performance based on F1-score metrics [18]. Feature-based methods such as TF–IDF are also commonly used in classifying Indonesian-language spam comments [19]. Recent work has further highlighted the effectiveness of supervised approaches for detecting spam that employs emoji or Unicode-based obfuscation [5], [2]. Beyond classical algorithms, ensemble methods that combine multiple learners have also been increasingly adopted to enhance spam-comment detection on social media platforms [4], [18]. For instance, a Random Forest ensemble was found to slightly outperform a single decision tree classifier (85.5% vs 84.4% accuracy) in an Indonesian classification task [20].

Nonetheless, studies that specifically focus on online-gambling promotional comments, especially on YouTube and within the Indonesian-language context, remain relatively limited. Most prior research tends to address generic spam classification or relies on English-language datasets [1]. Research on detecting Indonesian-language online gambling comments, particularly those that employ extreme character obfuscation using Unicode and emojis, has not been widely developed. This constitutes a research gap that must be addressed to curb the spread of online gambling content on YouTube more effectively. Given YouTube's continued, substantial growth in Indonesia, with an increasing number of active users each year, the problem is likely to worsen if left unaddressed. The prevalence of online-gambling promotional spam not only harms individual users but also adversely affects the nation's broader digital image. Consequently, technology-driven countermeasures, particularly machine learning

approaches, are increasingly important and relevant for maintaining a safe and positive digital ecosystem.

Furthermore, the increasing volume of spam comments on YouTube indicates that manual moderation is no longer sufficient to handle the scale of the problem. With more than 2.53 billion global users and 143 million users in Indonesia, the YouTube comment section has become a prime target for gambling promoters to disseminate illegal messages at scale. This surge in promotional comments not only disrupts user experience but also risks damaging the reputation of content creators, as their comment sections appear flooded with irrelevant and illegal content. Such fabricated interactions may also mislead YouTube's algorithm, since a high number of comments can be incorrectly interpreted as genuine engagement, even though many originate from gambling-promotion accounts. These conditions further reinforce the need for accurate and efficient automated solutions. Addressing this gap, the present study conducts a comparative evaluation of several widely used machine-learning algorithms—Multinomial Naive Bayes, Logistic Regression, Random Forest, and Support Vector Machine (SVM) [9] [8]—using an Indonesian-language YouTube comment dataset that contains diverse character-obfuscation patterns, including Unicode mixing, symbols, and emojis.

The primary objective of this research is to identify the most optimal and effective machine learning algorithm for detecting online-gambling promotional comments based on standard evaluation metrics: precision, recall, F1-score, and accuracy. The ultimate expectation is to produce a recommendation for the best-performing algorithm that can be implemented within YouTube's automated classification systems. In doing so, this work aims to help protect YouTube users from harmful content while contributing to subsequent research that may explore ensemble methods or deep learning as alternative approaches in the future.

METHODS

This study is systematically designed to compare the performance of four machine learning algorithms in detecting online-gambling promotional comments on YouTube. Figure 1 describes all detailed stages from data collection and preprocessing to model evaluation. Experiments were conducted in Python on Google Colab to ensure computational efficiency [21].

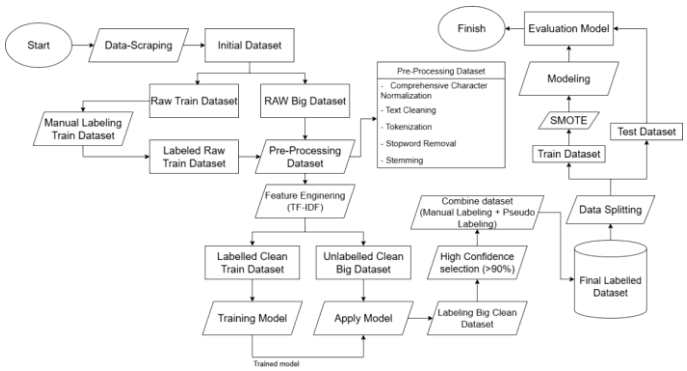


Figure 1. Research Workflow

Dataset Collection

Research data were collected from comments on YouTube videos uploaded to the channels of Deddy Corbuzier and Jess No Limit, spanning from 1 January 2025 to June 19, 2025, and covering a total of 160 videos. Retrieval utilised the YouTube Data API v3, the official method for accessing video comments legally and structurally [22].

Preprocessing

Preprocessing in this study introduces a novel character-normalization approach specifically designed to counter obfuscation strategies frequently used by online-gambling promoters. Unlike conventional text-cleaning pipelines, the proposed module employs a rule-based character map constructed through manual curation and AI-assisted pattern inspection of hundreds of Unicode variants, homoglyphs, emoji-styled letters, and decorative digits. This

mechanism consolidates visually manipulated expressions into canonical forms, enabling transformations such as "$l0t", "sl⌾t🎰", and "sl⭕t" into the readable token "slot", thereby recovering semantic integrity that is typically lost in standard preprocessing. After normalisation, regex-based cleaning, stopword removal, and Sastrawi stemming are applied before transforming tokens into TF–IDF n-gram representations. This preprocessing strategy ensures robustness against adversarial text manipulation and serves as a critical foundation for accurately detecting gambling-promotion comments.

Feature Engineering (TF-IDF)

Preprocessed text is transformed into numerical representations using Term Frequency–Inverse Document Frequency (TF-IDF).

Modelling and Pseudo-Labelling

Modelling in this study implements a semi-supervised pseudo-labelling approach to expand the training dataset [26] while maintaining annotation reliability. An initial set of 1,648 comments—balanced between 824 gambling-promotion and 824 non-gambling comments—was manually annotated and used to train the teacher model. The trained model was then applied to the unlabeled dataset, and only predictions with confidence scores above 0.90 were accepted as pseudo-labels and incorporated into the training set. This selective mechanism ensures that only high-certainty instances contribute to dataset enlargement, thereby preserving data quality and class representativeness in the context of gambling-promotion comment detection.

Data Splitting, Optimization, and Model Evaluation

The final labeled dataset is split using a stratified split into 80% training and 20% testing data. To address class imbalance, Synthetic Minority Oversampling Technique (SMOTE) is applied to the training set [23]. Next, hyperparameter optimization is performed for each algorithm using GridSearchCV to obtain the best configuration [24]. Final model performance is evaluated on the test set using accuracy, precision, recall, and f1-score.

RESULT AND DISCUSSION

This section presents the implementation outcomes and a comparative evaluation of four machine learning algorithms for detecting online-gambling promotional comments on YouTube. The experiments followed the methodology pipeline from data collection to final model evaluation on the test set.

Dataset Collection (Data Scraping)

Using the YouTube Data API v3, a total of 1,055,716 raw comments were collected. From these, 20,539 comments were selected as dataset candidates, and 1,648 comments were manually labelled into two classes—online-gambling promotion and non-gambling—to form the initial ground truth. Because full manual annotation was impractical, a pseudo-labelling approach was applied: an initial model trained on the labeled subset automatically assigned labels to the candidate set, and only high-confidence predictions were retained. This strategy effectively expanded the training data while reducing annotation effort, following adaptive-thresholding principles commonly used in semi-supervised text classification. Table 1 shows the collected comments.

Table 1. Dataset Types

video_id	video_channel	comment_author	comment_text
nQi59TFfQ1w	Deddy Corbuzier	@fetraarshanda	Bikin poling timnas om dedy
dwJrlxi4qtQ	Deddy Corbuzier	@SupendiAji	A*GU**S**T*OT***O*** aku gk boong, gmpang bgt WEDEY dsini
Qv7qu3ggFyY	Jess No Limit	@cengnunut6556	Meta ciki
8Ots7wzHNUs	Jess No Limit	@LexusRandolph-b7c	▷**SGI88** saya maen santai aja, tetep bisa WEDEY jutaan

Across the assembled dataset, this study focuses on a single primary feature, comment_text, for analysis. This selection is motivated by the fact that the comment text is the core representation of user communication in the dataset.

Dataset Preprocessing

The preprocessing pipeline aims to improve textual representations so that they are optimally interpretable by machine-learning algorithms. This is crucial given the unstructured nature of YouTube comments in our dataset and the frequent presence of non-standard text variants. Such variants include symbols, emojis, Unicode characters, and deliberate writing manipulations, e.g., character splitting or homoglyphs used to obscure gambling-related terms. Without appropriate normalization and transformation, a classifier would struggle to recognize patterns relevant to detection.

Accordingly, the preprocessing stages were carefully designed to handle these obfuscation forms effectively:

a. Comprehensive Character Normalization

We first normalize text using a custom character map that detects and converts non-standard characters to their canonical forms. The map covers hundreds of characters commonly used for obfuscation, such as Cyrillic letters (e.g., "C" from Russian, which resembles Latin "c"), mathematical/technical symbols (e.g., "***H***" → "h"), and alphanumeric-style emojis (e.g., "🅿" → "p"). The process goes beyond one-to-one substitutions by incorporating both linguistic and visual character logic, yielding more accurate and contextual normalization. After normalization, all text is lowercased (case folding) to avoid redundancy due to capitalization differences.

Table 2. Results of Comprehensive Normalization

Before Normalization	After Normalization
luar biasa gede nya 25:12 🅷🅺🅹🅴🅿🅴88	luar biasa gede nya 25:12 hkjepe88
🅿🆄🅻🅰🆄777, permainan seru dan penuh keuntungan!	pulau777 permainan seru dan penuh keuntungan!
*DORA*7**7** gachor ba*n*get, me***k***swin terus Busyet, ☆***ALEXIS***17 itu apaan sih? Kok heboh banget? sumpah?	dora77 gachor banget, mekswin terus busyet, alexis17 itu apaan sih? kok heboh banget? sumpah?

b. Text Cleaning

Next, we remove non-informative or noisy elements such as URLs, HTML tags, punctuation, and special symbols that do not contribute meaningfully to classification. Unlike standard practices that often delete digits, we retain numbers because they can signify key entities (e.g., gambling site names, referral codes, stake amounts). Cleaning is therefore performed selectively to preserve potentially important information.

Table 3. Text Cleaning Result

Before Cleaning	After Cleaning
luar biasa gede nya 25:12 hkjepe88	luar biasa gede nya2512 hkjepe88
pulau777 permainan seru dan penuh keuntungan!	pulau777 permainan seru dan penuh keuntungan
dora77 gachor banget, mekswin terus busyet, alexis17 itu apaan sih? kok heboh banget? sumpah?	dora77 gachor banget mekswin terus busyet alexis17 itu apaan sih kok heboh banget sumpah

c. Tokenization
After cleaning, we perform tokenization, i.e., splitting each string into word-level tokens. This enables word-level analysis during learning and facilitates subsequent transformations such as stopword removal and stemming.

Table 4. Tokenization

Before Tokenization	After Tokenization
luar biasa gede nya2512 hkjepe88	"luar","biasa","gede","nya2512","hkjepe88"
pulau777 permainan seru dan penuh keuntungan	"pulau777","permainan","seru","dan","penuh","keuntungan"
dora77 gachor banget mekswin terus	"dora77","gachor","banget","mekswin","terus"
busyet alexis17 itu apaan sih kok heboh banget sumpah	"busyet","alexis17","itu","apaan","sih","kok","heboh","banget","sumpah"

d. Stopword Removal
We eliminate common Indonesian function words (e.g., yang, dari, dan, di) that carry little discriminative power for this task. Removing stop words helps the model focus on more informative features.

e. Stemming
Finally, tokens are reduced to their base forms using the Nazief–Adriani algorithm via the Sastrawi library, which is well-established for Indonesian. This consolidates morphological variants (e.g., bermain, dimainkan, pemain → main), improving learning efficiency and model generalization.

Feature Engineering (TF-IDF)

TF–IDF assigns word weights by evaluating each term's relevance within a document relative to the entire corpus. Terms with high d iscriminative value (e.g., "deposit," "jackpot") receive higher weights. TF–IDF is computed as follows:

$$TF - IDF(t,d) = TF(t,d) \times IDF(t) \quad (1)$$

With

$$TF(t,d) = \frac{f_{t,d}}{\sum_k \quad f_{k,d}} \quad (2)$$

$$IDF(t) = log\frac{N}{1+n_t} \quad (3)$$

Here $f_{t,d}$ is the frequency of term t in document d;$\sum_k \quad f_{k,d}$ is the total number of terms in d; N is the total number of documents in the corpus; and n_t is the number of documents containing t. In this study, TF–IDF is applied after preprocessing, retaining the most relevant, high-weight terms while constraining feature dimensionality to reduce overfitting.

Modelling and Pseudo-Labelling

The first step is training a teacher model on the 1,648 manually labeled instances. Four algorithms are evaluated for this role: Multinomial Naive Bayes (MNB), Logistic Regression (LR), Random Forest (RF), and Support Vector Machine (SVM). The model with the highest cross-validated F1-score is selected for the next stage. The teacher model then predicts labels for the 20,539 unlabeled instances. Predictions with a confidence score > 90% are accepted as pseudo labels and merged with the manually labeled data, yielding the final labeled dataset.

Data Splitting and Model Optimisation

After pseudo-labelling, the final labelled dataset comprises 9,111 rows: 5,570 non-promotion comments (label 0) and 3,541 gambling-promotion comments (label 1). The dataset is split using a stratified split into 80% training and 20% testing to preserve class proportions. Because the class distribution remains imbalanced, the Synthetic Minority Over-sampling Technique (SMOTE) is applied to the training set. After SMOTE, the training data became

balanced, with 8,912 samples, 4,456 per class, which helped prevent bias toward the majority class.

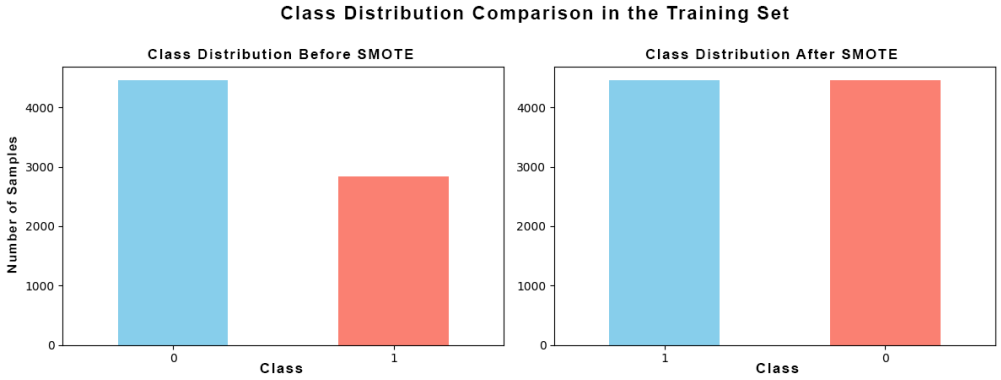


Figure 2. Class Distribution Before and After SMOTE

Next, hyperparameter optimization is conducted for MNB, LR, RF, and SVM using GridSearchCV with 5-fold cross-validation, using macro F1-score as the primary metric. The best parameters for each model are then used for final training. Table 5 summarizes the search space and the optimal parameter combinations obtained.

Table 5. Hyperparameter Optimization

Model	Tuned Parameter(s)	Parameter Search Space	Best Parameter(s)
MNB	alpha	[0.01, 0.1, 0.5, 1.0]	alpha=0.1
LR	C, solver	C=[0.1, 1, 10, 100], solver=['liblinear']	C=10, solver='liblinear'
RF	n_estimators, max_depth	n_estimators=[100, 200], max_depth=[20,50, None]	n_estimators=100, max_depth=None
SVM	C, kernel, gamma	C=[1, 10, 100], kernel=['rbf'], gamma=['scale', 'auto']	C=1, kernel='rbf', gamma='scale'

Model Evaluation

Evaluation is performed on the test set of 1,823 comments, 1,114 non-promotion, and 709 promotion across four metrics: accuracy, precision, recall, and F1-score for each class, with a particular focus on minority-class performance (promotion/label 1). Table 6 presents the summary of results for all four models.

Table 6. Model Evaluation Results

Model	Accuracy	Precision (Promotion)	Recall (Promotion)	F1-Score (Promotion)
MNB	0.9879	0.9942	0.9746	0.9843
LR	0.9923	0.9915	0.9887	0.9901
RF	0.9923	0.9971	0.9831	0.9900
SVM	**0.9929**	**0.9957**	**0.9859**	**0.9908**

The model performance on the minority class (gambling promotion) remains highly stable, with precision above 0.991 and recall at least 0.974, indicating that most promotional comments—including those obfuscated with Unicode variants, symbols, or emojis—were successfully detected. The consistently high F1-scores (>0.98) demonstrate the effectiveness of the customized preprocessing pipeline in normalizing obfuscated characters into canonical forms, enabling clearer feature representation. Additionally, SMOTE mitigated class imbalance by generating representative minority samples, contributing to the strong recall scores across models. Together, these components enabled all classifiers—particularly SVM—to achieve robust and balanced performance.

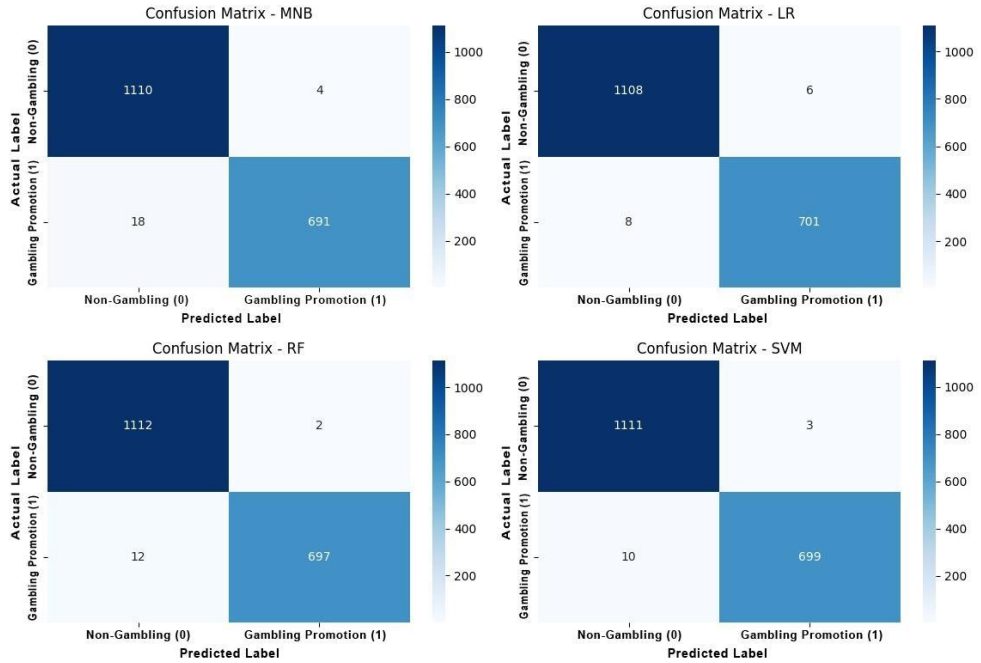


Figure 3. Confusion Matrices

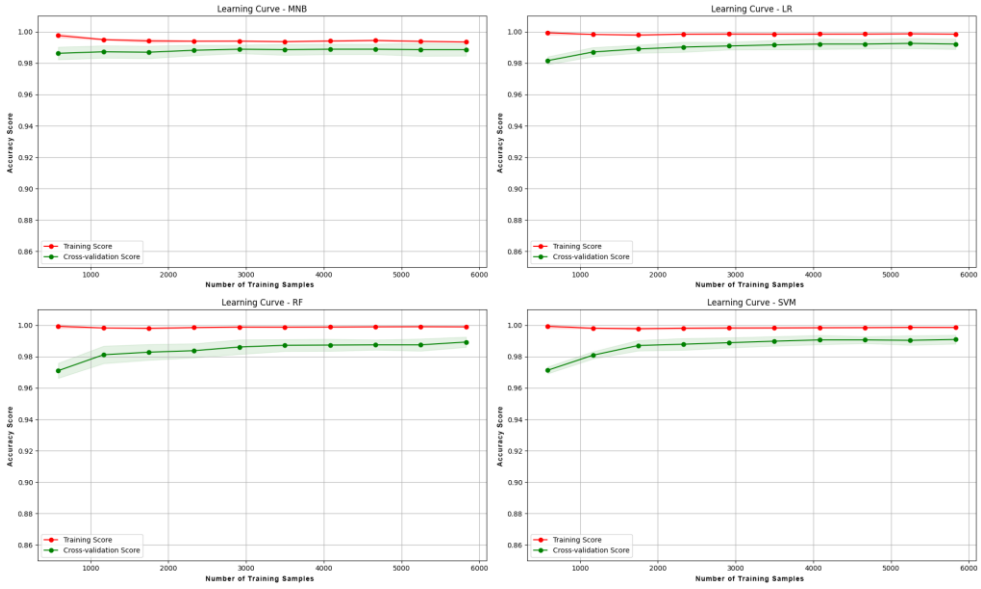


Figure 4. Learning Curves

Analysis and Interpretation

The evaluation results show that all algorithms achieve consistently high accuracy in classifying online-gambling promotional comments. Multinomial Naive Bayes (MNB) delivers strong performance with a recall of 0.9746, enabling effective identification of true positive promotion cases. Random Forest (RF) and SVM demonstrate higher precision, indicating stronger confidence in predicting promotion-class comments with minimal false positives, while SVM offers the most balanced performance and maintains stable learning-curve behavior as training size increases.

Figure 5 illustrates the boxplot distribution of F1-Macro scores across the four models under 5-fold cross-validation. The distributions are tightly concentrated in the 0.98–1.00 range, reflecting high consistency across folds. Random Forest exhibits slightly wider variability due to a small number of lower outliers, whereas Logistic Regression and SVM present the most compact distributions. This visual pattern aligns with the ANOVA result ($p = 0.567$),

confirming that the differences in mean performance among the models are not statistically significant.

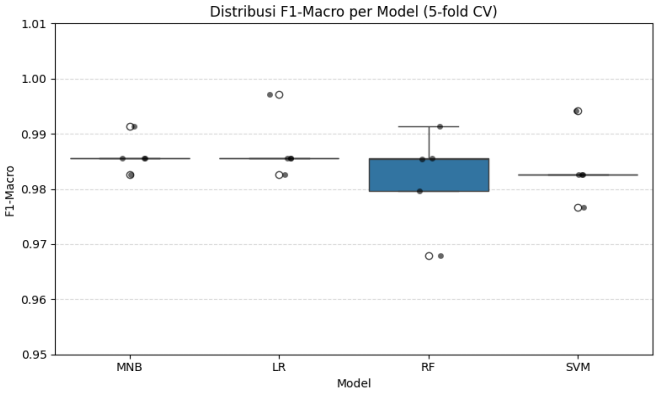


Figure 5. Boxplot of F1-Macro scores across the four models under 5-fold cross-validation.

A baseline version of the four models—trained only on the manually labeled dataset without pseudo-labelling and SMOTE—was evaluated to measure the effect of the optimization techniques. As shown in Table 7, baseline performance decreases considerably for the minority (promotion) class: MNB_base and LR_base record F1-scores of 0.9373 and 0.9201, RF_base drops to 0.9000, and SVM_base declines to 0.9269 due to class imbalance. In contrast, all optimiszed models surpass an F1 score of 0.98. These results demonstrate that pseudo-labelling effectively expands the labelled dataset and that SMOTE rebalances the training distribution, jointly producing a substantial improvement in detecting gambling-promotion comments, particularly in the minority class.

Table 7. Baseline Model Performance (Without Pseudo-Labelling and SMOTE)

Model	Accuracy	Precision (Promotion)	Recall (Promotion)	F1-Score (Promotion)
MNB_base	0.9363	0.9235	0.9515	0.9373
LR_base	0.9242	0.9729	0.8727	0.9201
RF_base	0.9030	0.9290	0.8727	0.9000
SVM_base	0.9303	0.9733	0.8848	0.9269

The strong performance of the classifier is supported by four major components:

1. Adaptive preprocessing that addresses character manipulations such as emojis, Unicode characters, irregular spacing, and letter scrambling.
2. TF–IDF vectorization that highlights key terms for detecting gambling promotions (e.g., “bonus,” “daftar,” “slot,” “jackpot”).
3. Pseudo-labelling, which effectively expands the labelled set from 1,648 to 9,111 comments without exhaustive manual annotation.
4. SMOTE to balance the training distribution, enabling fair learning for both classes.

Several challenges were encountered during the experimentation process, including severe class imbalance prior to SMOTE and substantial variation in character manipulation strategies used by promoters, which made preprocessing essential for consistent tokenization. Additionally, since the dataset was collected from only two YouTube channels, there remains a potential limitation regarding the generalization of model performance to other content categories. Nonetheless, the combined pipeline of preprocessing, semi-supervised learning, and algorithm optimization provides an accurate and efficient detection framework for identifying gambling-promotional content in YouTube comments.

CONCLUSION AND RECOMMENDATION

This study designs an automated detector for online-gambling promotional comments on YouTube using a machine-learning pipeline augmented by semi-supervised pseudo-

labeling. In line with the research objective of selecting an effective classifier, the approach expands the labelled set from 1,648 to 9,111 comments without additional manual annotation. A tailored preprocessing stack (comprehensive normalization of Unicode/emoji/irregular spacing, stopword removal, and Nazief–Adriani stemming via Sastrawi) combined with TF–IDF features yields high-quality text representations.

Across four models, Multinomial Naive Bayes, Logistic Regression, Random Forest, and Support Vector Machine all achieve >98% accuracy on the test set. SVM delivers the most balanced performance, with an F1 score of 0.9908 on the promotion class, making it the recommended choice for automated moderation. Confusion matrices indicate very low misclassification, and learning curves are stable, suggesting neither underfitting nor overfitting. Nevertheless, this study has limitations, chiefly the scope of algorithms evaluated restricted to classical supervised models. Accordingly, future work should explore more advanced ensemble approaches—such as gradient boosting, XGBoost, or stacking classifiers—which can combine multiple weak learners to improve robustness against diverse obfuscation styles and reduce sensitivity to noisy textual variations. In addition, deep learning architectures such as BiLSTM, CNN-based text classifiers, or transformer-based models (e.g., IndoBERT) may offer stronger z by learning contextual and subword-level patterns that classical models cannot capture. These models could be particularly effective in detecting more extreme obfuscation strategies, adapting to broader Youtube comment domains, and handling complex linguistic distortions commonly used in gambling-promotion comments.

REFERENCES

[1] R. B. Perdana, Ardin, I. Budi, A. B. Santoso, A. Ramadiah, and P. K. Putra, "Detecting Online Gambling Promotions on Indonesian Twitter Using Text Mining Algorithm," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 8, pp. 942–949, 2024, doi: https://doi.org/10.14569/IJACSA.2024.0150893

[2] A. Sakpal, "A Survey on Emoji and Unicode-Based Data Masking Attacks On Ai Systems," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 13, no. 6, pp. 1805–1810, 2025, Doi: https://doi.org/10.22214/ijraset.2025.72523

[3] Y. Zhou, Y. Xiao, W. Ai, and G. Gao, "The Hidden Language Of Harm: Examining The Role Of Emojis In Harmful Online Communication And Content Moderation," 2025, doi: https://doi.org/10.48550/arXiv.2506.00583

[4] H. Oh, "A Youtube Spam Comments Detection Scheme Using Cascaded Ensemble Machine Learning Model," *IEEE Access*, vol. 9, no. 04, pp. 144121–144128, 2021, doi: https://doi.org/10.1109/ACCESS.2021.3121508

[5] B. Sai, S. Vandana, K. Sreeja, N. Bhavana, and K. G. Babu, "Spam Comment Detection on Social Media A Hybrid Approach with Emoji Feature, Post-Comment Pairs And Ensemble Machine Learning," pp. 2156–2159, 2025.

[6] D. J. W. Ikram and S. F. Pane, "Deteksi Spam Bot Pada Komentar Youtube : Tinjauan Literatur Sistematis," *CSRID Journal*, vol. 15, no. 2, pp. 103-123, 2023, doi: https://www.doi.org/10.22303/csrid.15.2.2022.103-123

[7] R. Abdillah, I. Adkha, D. P. Agustin, and N. Alam, "Sosialisasi Penerapan Algoritma Media Sosial Youtube Untuk Menaikkan Jumlah Pengunjung," *Karunia Jurnal Has. Pengabdi. Masy. Indones.*, vol. 4, no. 1, pp. 120–130, 2025, doi: https://doi.org/10.58192/karunia.v4i1.3075

[8] P. Roshini and B. Indira, "Spam Detection for Youtube Comments Using Machine Learning Algorithms," *International Journal of Current Science*, vol. 12, no. 4, pp. 394-397, 2022.

[9] A. S. Xiao and Q. Liang, "Spam Detection for Youtube Video Comments Using Machine Learning Approaches," *Mach. Learn. With Appl.*, vol. 16, pp. 100550, 2024, doi: https://doi.org/10.1016/j.mlwa.2024.100550

[10] M. Tsiourlini, K. Tzafilkou, D. Karapiperis, and C. Tjortjis, "Text Analytics on Youtube Comments For Food Products," *Information*, vol. 15, no. 10, 2024, doi: https://doi.org/10.3390/info15100599
[11] M. Afrad, D. C. Febrianto, S. Wijayanto, and M. Y. Fathoni, "Sentiment Analysis of Visitor Reviews on Baturaden Tourist Attraction Using Machine Learning Methods," *Edu Komputika J.*, vol. 11, no. 1, pp. 57–64, 2024, doi: https://doi.org/10.15294/edukom.v11i1.10561
[12] K. D. A. Salsabila and N. Trianasari, "Analisis Persepsi Produk Kosmetik Menggunakan Metode Sentiment Analysis Dan Topic Modeling (Studi Kasus: Laneige Water Sleeping Mask)," *Jurnal Teknologi dan Manajemen Informatika*, vol. 7, no. 1, pp. 1–9, 2021, doi: https://doi.org/10.26905/jtmi.v7i1.5593
[13] G. I. M. Putra, M. S. Riyadi, A. Maulana, and S. Maesaroh, "Analysis of The Application of Machine Learning Algorithm in Spam Detection System: Literature Review," *J. Artif. Intell. Eng. Appl.*, vol. 4, no. 3, pp. 1615–1621, 2025, doi: https://doi.org/10.59934/jaiea.v4i3.965
[14] A. Sinhal and M. Maheshwari, "An Extensive Review on Contemporary Analysis of Comment Filtration of Youtube Videos Using Machine Learning Techniques," *Int. J. Emerg. Technol. Adv. Eng.*, vol. 12, no. 9, pp. 130–141, 2022, doi: https://doi.org/10.46338/ijetae0922_14
[15] K. Li, "Analysis of Spam Classification Based on Naive Bayes And Random Forest Model," *Adv. Econ. Manag. Polit. Sci.*, vol. 84, no. 1, pp. 250–257, 2024, doi: https://doi.org/10.54254/2754-1169/84/20240817
[16] A. Dewandaru and J. S. Wibowo, "Analisis Sentimen Dan Klasifikasi Tweet Terkait Mutasi Covid-19 Menggunakan Metode Naïve Bayes Classifier," *Jurnal Teknologi dan Manajemen Informatika*, vol. 8, no. 1, pp. 32–38, 2022, doi: https://doi.org/10.26905/jtmi.v8i1.6803
[17] N. Venkatramana, C. Jashnavi, J. J. Guptha, P. C. Sekhar, and M. Chandra, "Random Tree Classifier: A Machine Learning Spam Comment Detection on Youtube," *International Journal of Scientific Research in Science and Technology*, vol. 11, no. 11, pp. 311–317, 2024, doi: https://ijsrst.com/IJSRST24111146
[18] G. Airlangga, "Spam Detection on Youtube Comments Using Advanced Machine Learning Models: A Comparative Study," *Brill. Res. Artif. Intell.*, vol. 4, no. 2, pp. 500–508, 2024, doi: https://doi.org/10.47709/brilliance.v4i2.4670
[19] A. N. Anggraeni, K. Mustofa, and S. Priyanta, "Comparison of Filter And Wrapper Based Feature Selection Methods On Spam Comment Classification," *Ijccs (Indonesian J. Comput. Cybern. Syst.*, vol. 15, no. 3, pp. 245-254, 2021, doi: https://doi.org/10.22146/ijccs.66965
[20] F. Y. Pamuji and V. P. Ramadhan, "Komparasi Algoritma Random Forest Dan Decision Tree Untuk Memprediksi Keberhasilan Immunotheraphy," *Jurnal Teknologi Dan Manajemen Informatika*, vol. 7, no. 1, pp. 46–50, 2021, doi: https://doi.org/10.26905/jtmi.v7i1.5982
[21] S. Balaraman, "Comparison Of Classification Models For Breast Cancer Identification Using Google Colab," *Preprints.*, pp. 1–11, 2020, doi: https://doi.org/10.20944/preprints202005.0328.v1
[22] A. Deubel, J. Breuer, J. Kohne, and M. R. Mohseni, "Overview of Working With Data From Youtube," pp. 1–17, 2024, doi: 10.60762/Ggdbd24012.1.0
[23] A. Glazkova, "A Comparison of Synthetic Oversampling Methods For Multi-Class Text Classification," No. 18, pp. 1–12, 2020, doi: https://doi.org/10.48550/arXiv.2008.04636
[24] D. Budiman, Z. Zayyan, A. Mardiana, and A. A. Mahrani, "Email Spam Detection: A Comparison of Svm And Naive Bayes Using Bayesian Optimization And Grid Search Parameters," *J. Student Res. Explor.*, vol. 2, no. 1, pp. 53–64, 2024, doi: https://doi.org/10.52465/josre.v2i1.260