International Journal of Engineering, Science and Information Technology
Volume 6, No. 1 (2026) pp. 1-8
eISSN: 2775-2674
Website: http://ijesty.org/index.php/ijesty
DOI: https://doi.org/10.52088/ijesty.v6i1.1794
**Research Paper, Short Communication, Review, Technical Paper**

# Comparison of Support Vector Machine and Naïve Bayes Algorithms Based on TF-IDF in Online Gambling Website Detection

**Rina Refianti***, **Husein Alhafiz**

*Faculty of Computer Science and Information Technology, Gunadarma University, Jakarta, Indonesia*

**Corresponding author Email: rina@staff.gunadarma.ac.id*

*The manuscript was received on 22 July 2025, revised on 28 August 2025, and accepted on 28 December 2025, date of publication 20 January 2026*

**Abstract**

The rapid growth of digital technology has significantly accelerated the spread of illegal online content, particularly gambling websites, which threaten social stability and regulatory enforcement. To address this issue, this study develops an automated detection system for online gambling sites using text classification with the Term Frequency–Inverse Document Frequency (TF-IDF) approach. A total of 1,225 website URLs were collected through web scraping, and after preprocessing, 1,166 valid entries were manually labeled into two classes: gambling and normal. The preprocessing steps included cleaning, tokenizing, stopword removal, stemming, and domain parsing, followed by feature extraction using TF-IDF, which generated 2,426 numerical features. To mitigate class imbalance, the Synthetic Minority Oversampling Technique (SMOTE) was applied to the training dataset. Two machine learning algorithms were implemented and compared: Support Vector Machine (SVM) with multiple kernels (Linear, RBF, Polynomial, and Sigmoid) and Multinomial Naïve Bayes (MNB). Experimental evaluation was conducted using accuracy, precision, recall, specificity, and F1-score metrics. Results demonstrate that SVM with the RBF kernel achieved the best performance, with an accuracy of 91.88% and an F1-score of 93.70%, while MNB obtained an accuracy of 88.46% and an F1-score of 91.00%. These findings confirm that SVM, particularly with the RBF kernel, delivers more stable and accurate performance in distinguishing gambling websites from normal ones. The proposed system offers a reliable foundation for the development of automated tools to monitor, detect, and block illegal online gambling content, thereby supporting regulatory enforcement and reducing the negative societal impacts of online gambling.

**Keywords**: *Multinomial Naïve Bayes, Online Gambling Detection, Support Vector Machine, Text Classification, TF-IDF.*

## 1. Introduction

The rapid development of digital technology has accelerated the spread of illegal online content, including gambling websites. These sites are considered cybercrimes and pose serious risks to society, ranging from financial losses to social disruption [1]. In Indonesia, the proliferation of online gambling has become a critical concern, requiring not only legal action but also technological solutions to support prevention efforts [2] .

Various approaches have been proposed to detect malicious websites, such as blacklist-based, URL-based, content-based, and hybrid methods. Blacklist approaches are easy to implement but fail to detect new domains [3], while URL-based techniques rely on lexical features that often lack sufficient accuracy[4]. Content-based detection, particularly text classification, is widely used to capture semantic patterns from website content and has shown more promising results [5].

Machine learning methods, especially Support Vector Machine (SVM) and Naïve Bayes (NB), have been extensively applied in text classification tasks. SVM is highly effective in handling high-dimensional features generated by TF-IDF, while NB provides a simple and efficient probabilistic framework [6][7]. Comparative studies indicate that SVM often yields higher accuracy, whereas NB offers faster computation and scalability [8].

A key challenge in online gambling site detection is class imbalance, where the number of gambling URLs is significantly lower than that of normal sites. To overcome this, the Synthetic Minority Oversampling Technique (SMOTE) can be applied to rebalance the dataset and enhance classifier performance [9]. In addition, preprocessing techniques—such as cleaning, tokenization, stopword removal, stemming, and TF-IDF transformation—are crucial for converting raw textual data into meaningful numerical features for classification [10].

Further improvements can be achieved by combining preprocessing with effective feature engineering. Proper data cleansing and normalization enhance classification results [10], while Natural Language Processing (NLP) methods provide broader adaptability for analyzing online content [11]. Previous studies also highlight that optimization of SVM using grid search and n-gram features significantly improves text classification performance. At the same time, class imbalance challenges have been effectively addressed through SMOTE, which strengthens model robustness [16]. Moreover, Naïve Bayes has proven to be an efficient baseline method for sentiment and text classification due to its simplicity and computational efficiency [13].

## 2. Literature Review

### 2.1. Online Gambling Detection

Online gambling is one of the most prevalent forms of cybercrime that threatens social and economic stability. In Indonesia, online gambling websites are illegal and subject to strict law enforcement [2]. Various approaches have been proposed to detect malicious websites, such as blacklist-based, URL-based, and content-based methods. Blacklist approaches are easy to apply but cannot detect new domains[14], while URL-based methods rely on lexical features that are often insufficient [15]. Content-based detection, particularly using text classification, has gained more attention due to its ability to capture semantic patterns from website content [11].

### 2.2. Text Classification

Text classification is a core task in Natural Language Processing (NLP) that involves assigning categories to textual data based on its content. Techniques such as Term Frequency–Inverse Document Frequency (TF-IDF) are widely used to convert text into numerical features [15]. In recent works, text classification performance has been further enhanced using deep contextual models such as BERT, which provide richer semantic understanding in hoax detection[16]. This shows that combining traditional approaches with modern NLP methods can lead to significant improvements in classification results.

### 2.3. Support Vector Machine (SVM)

Support Vector Machine (SVM) is a supervised learning algorithm that constructs an optimal hyperplane to separate data points into classes. SVM is widely used in text classification due to its ability to handle high-dimensional data and achieve high accuracy [6]. Several kernel functions can be applied, such as Linear, Radial Basis Function (RBF), Polynomial, and Sigmoid, to map features into higher dimensions and improve performance [7]. Studies have also compared kernel performance, where RBF and linear kernels showed varying effectiveness depending on the dataset, indicating that kernel selection plays an essential role in optimizing classification results [17].

### 2.4. Multinomial Naïve Bayes

Naïve Bayes (NB) is a probabilistic classifier based on Bayes' theorem, assuming feature independence. Research applying Naïve Bayess effective for text classification because of its efficiency and relatively good performance in handling large datasets [8]. The Multinomial Naïve Bayes (MNB) variant is particularly suited for modeling term frequencies in documents, making it a strong baseline for text-based applications [18]. Research applying Naïve Bayes for sentiment analysis on user reviews also demonstrates its adaptability in real-world classification tasks, further reinforcing its importance in text classification studies [19] .

### 2.5. Handling Imbalanced Data

A common challenge in online gambling detection is dataset imbalance, where gambling site samples are significantly fewer than normal site samples. This imbalance can bias machine learning models toward the majority class, reducing their ability to detect minority instances. The Synthetic Minority Oversampling Technique (SMOTE) has been widely adopted to address this issue by generating synthetic samples for the minority class, thus improving model performance and generalization [9][20]. Other works also confirmed that applying SMOTE in text classification leads to better balance between classes and improved predictive performance. Furthermore, integrating SMOTE with advanced classification models such as Random Forest and SVM has been shown to achieve higher accuracy in emotion classification and sentiment prediction tasks [21].

## 3. Methods

The research methodology applied in this study is illustrated in Figure 1, which shows the overall stages from data collection to evaluation.

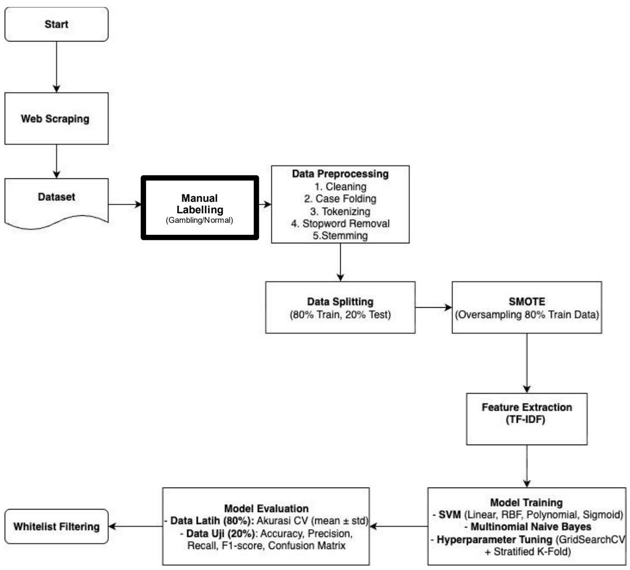


**Fig 1.** Research Methodology of Online Gambling Website Detection

### 3.1. Data Collection

The dataset used in this study was obtained through web scraping using the SerpAPI tool. A total of 1,225 website URLs were collected from multiple sources and categorized into two groups: online gambling sites and normal sites. Manual labeling was performed to ensure data validity and served as the ground truth for training and testing the models. The research methodology applied in this study is illustrated in Figure 1, which shows the overall stages from data collection to evaluation.

### 3.2. Dataset Labelling

Since supervised learning requires labeled data, each website was manually inspected and assigned a label: "Judi" for online gambling and "normal" for normal sites. This process ensured that the dataset could be used to train and evaluate classification models effectively.

### 3.3. Data Preprocessing

Data preprocessing was performed to prepare the collected text before feature extraction and classification. The main steps include:

a. Cleaning: Removing irrelevant characters such as numbers, punctuation, symbols, and duplicate entries from page titles.
b. Case Folding: Converting all characters into lowercase to ensure text uniformity.
c. Tokenization: Splitting sentences into individual words (tokens).
d. Stopword Removal: Eliminating common words with little semantic value.
e. Stemming: Reducing words to their root forms to standardize variations.

Domain Extraction: Extracting domain names from URLs and combining them with preprocessed titles to form the final text input.

This preprocessing ensured the dataset was clean, consistent, and suitable for feature extraction using TF-IDF.

### 3.4. Dataset Splitting

The dataset was divided into two subsets using an 80:20 ratio with stratified sampling to preserve class distribution. The training set (80%) was used for model training and cross-validation, while the testing set (20%) was reserved for final performance evaluation. To address class imbalance, the Synthetic Minority Oversampling Technique (SMOTE) was applied only to the training set after TF-IDF transformation, ensuring no data leakage to the testing set.

### 3.5. Model Training

Model training is the stage where the classifiers are trained using the training set and evaluated to identify the best-performing configuration. In this study, two algorithms were implemented: Support Vector Machine (SVM) and Multinomial Naïve Bayes (MNB), with TF-IDF feature representations of the text.

For the SVM algorithm, the model constructs an optimal hyperplane to separate the classes. Several kernel functions were tested, including Linear, Polynomial, Radial Basis Function (RBF), and Sigmoid, to examine their impact on classification performance. Hyperparameter tuning was performed using GridSearchCV to select the best parameter settings. The evaluation on the training set was conducted using Stratified K-Fold Cross Validation (k=5), which ensures that each fold maintains the class distribution and provides more reliable performance estimation.

For the MNB algorithm, Laplace smoothing with parameter $\alpha = 1.0$ was applied, which is effective in handling zero-frequency problems in text data. Due to its simpler probabilistic nature, MNB was trained on the training set and directly tested on the unseen test set without cross-validation.

Both models were trained on TF-IDF-transformed text, and their performance was later compared based on evaluation metrics to determine the most effective algorithm for online gambling website detection.

### 3.6. Evaluation

Evaluation is the final stage of this study to measure the performance of classification models in detecting online gambling websites. This stage describes the computational process applied to the models, datasets, and algorithms that have been defined previously. The evaluation results are presented using a confusion matrix consisting of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). Furthermore, performance is measured using accuracy, precision, recall, and F1-score to provide a comprehensive assessment of the classifiers.

## 4. Results and Discussion

### 4.1. Dataset Labelling Result

The author labeled the dataset that had been obtained from the data collection stage by examining both the URL and page title of each website. The labeling process was carried out manually to ensure data validity, where websites containing gambling-related content were assigned the gambling label, while other websites were categorized as normal. After this process, the author obtained 769 data labeled as gambling and 456 data labeled as normal, resulting in a total of 1,225 entries. The distribution of the dataset for each category is presented in Table 1 and visualized in Table 1.

**Table 1.** Dataset labeling result

| No. | Label | | Amount of Data |
|---|---|---|---|
| 1. | Gambling | | 769 |
| 2. | Normal | | 456 |
| | | Total | 1225 |

### 4.2. Model Training Result

During the training phase, a total of 1,166 labeled website URLs were used after preprocessing. The dataset was split into 80% training and 20% testing, and the Synthetic Minority Oversampling Technique (SMOTE) was applied to the training set to balance the gambling and normal classes. Feature extraction using TF-IDF produced 2,426 numerical features for classification.

Support Vector Machine (SVM) was trained with four kernel types—Linear, Radial Basis Function (RBF), Polynomial, and Sigmoid—using Grid Search for hyperparameter tuning. Multinomial Naïve Bayes (MNB) was trained using default parameters as a baseline. The training process indicated that SVM required longer computation due to kernel optimization, while MNB was faster and more efficient. However, SVM models, particularly with the RBF kernel, consistently showed more stable convergence and better adaptability to the feature space.

### 4.3. Evaluation Result

The models that had been trained and validated were subsequently tested on the testing dataset, and the results were stored in CSV format. To evaluate model performance, confusion matrices were generated for each classifier to visualize the distribution of predictions across classes. The confusion matrices of the five models (SVM with Linear, RBF, Polynomial, and Sigmoid kernels, and Multinomial Naïve Bayes) are shown in Figure 2a–2e.

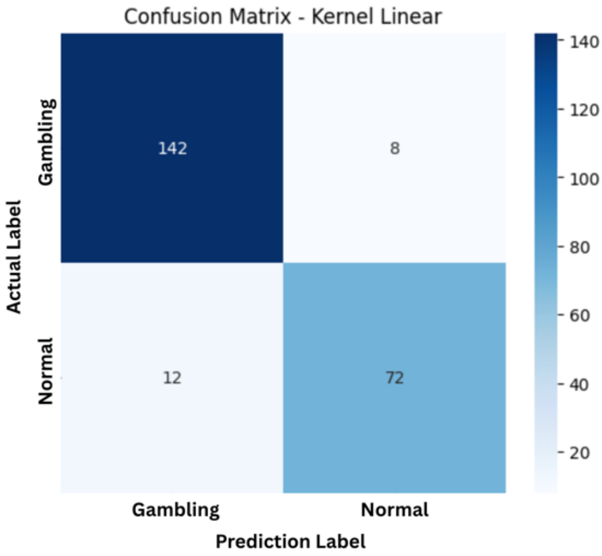


**Fig 2a.** Confusion matrix of SVM Linear kernel

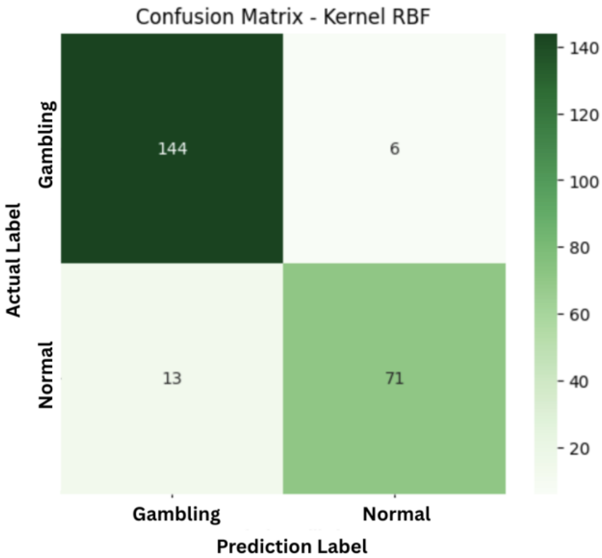


**Fig 2b.** Confusion matrix of SVM RBF kernel

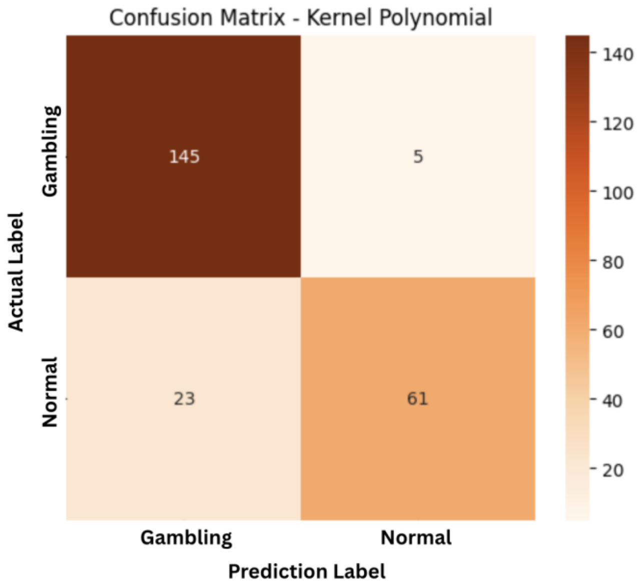


**Fig 2c.** Confusion matrix of SVM Polynomial kernel

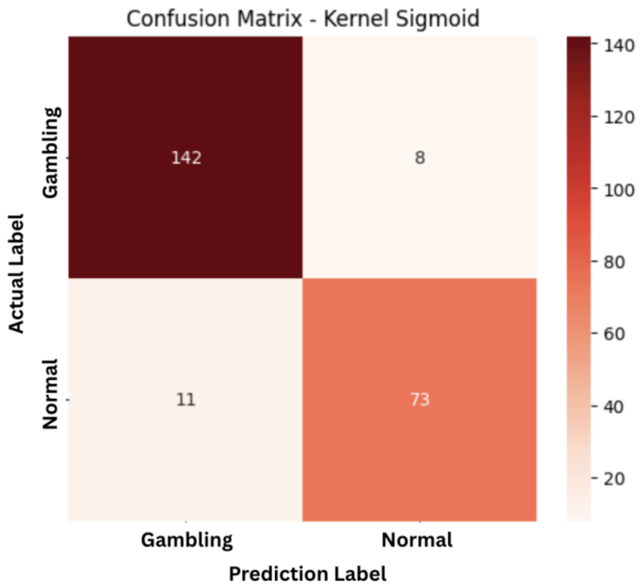


**Fig 2d.** Confusion matrix of SVM Sigmoid kernel

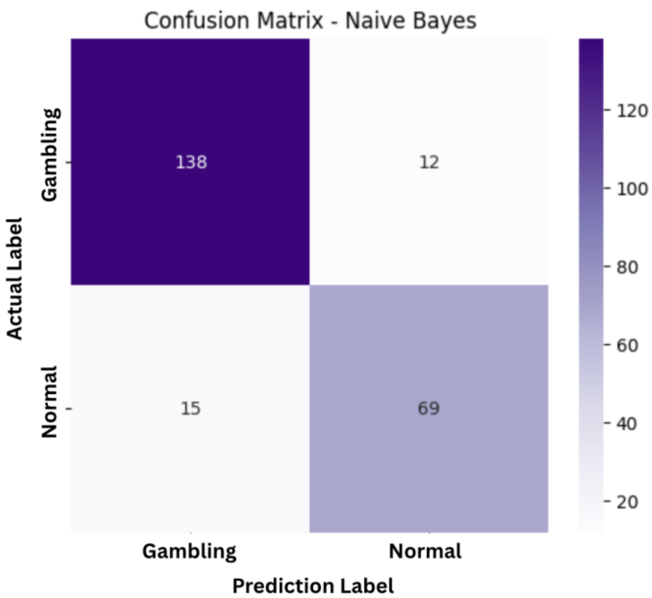


**Fig 2e.** Confusion matrix of Multinomial Naïve Bayes

Based on Figure 2a–2e, the SVM Linear kernel produced the best balance between true positive and true negative predictions, while other kernels such as Polynomial and Sigmoid showed more misclassifications. Multinomial Naïve Bayes, on the other hand, achieved relatively stable performance but was slightly less accurate than SVM Linear.

After obtaining the confusion matrix results, the evaluation was continued using the classification_report() function from the Scikit-learn library to calculate the accuracy, precision, recall, and F1-score of each model. The classification reports are presented in Figure 3a–3e.

--- Evaluation of Linear Kernel Model ---
Accuracy: 91.45%

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| gambling | 0.92 | 0.95 | 0.93 | 150 |
| normal | 0.90 | 0.86 | 0.88 | 84 |
| accuracy | | | 0.91 | 234 |
| macro avg | 0.91 | 0.90 | 0.91 | 234 |
| weighted avg | 0.91 | 0.91 | 0.91 | 234 |

**Fig 3a.** Classification report of SVM Linear kernel

--- Evaluaton of Model Kernel RBF ---
Accuracy: 91.88%

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| gambling | 0.92 | 0.96 | 0.94 | 150 |
| normal | 0.92 | 0.85 | 0.88 | 84 |
| accuracy | | | 0.92 | 234 |
| macro avg | 0.92 | 0.90 | 0.91 | 234 |
| weighted avg | 0.92 | 0.92 | 0.92 | 234 |

**Fig 3b.** Classification report of SVM RBF kernel

--- Evaluation of Polynomial Kernel Model ---
Accuracy: 88.03%

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| gambling | 0.86 | 0.97 | 0.91 | 150 |
| normal | 0.92 | 0.73 | 0.81 | 84 |
| accuracy | | | 0.88 | 234 |
| macro avg | 0.89 | 0.85 | 0.86 | 234 |
| weighted avg | 0.89 | 0.88 | 0.88 | 234 |

**Fig 3c.** Classification report of SVM Polynomial kernel

--- Evaluation of Sigmoid Kernel Model ---
Accuracy: 91.88%

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| gambling | 0.93 | 0.95 | 0.94 | 150 |
| normal | 0.90 | 0.87 | 0.88 | 84 |
| accuracy | | | 0.92 | 234 |
| macro avg | 0.91 | 0.91 | 0.91 | 234 |
| weighted avg | 0.92 | 0.92 | 0.92 | 234 |

**Fig 3d.** Classification report of SVM Sigmoid kernel

```
--- Evaluation of Naive Bayes Model ---
Accuracy: 88.46%

              precision    recall  f1-score   support

    gambling       0.90      0.92      0.91       150
      normal       0.85      0.82      0.84        84

    accuracy                           0.88       234
   macro avg       0.88      0.87      0.87       234
weighted avg       0.88      0.88      0.88       234
```

**Fig 3e.** Classification report of Multinomial Naïve Bayes

As shown in Figure 3a–3e, the SVM Linear kernel achieved the highest accuracy and F1-score, demonstrating its superiority for text classification in online gambling detection. The RBF kernel performed slightly lower but remained competitive. Polynomial and Sigmoid kernels exhibited weaker results due to their higher misclassification rates, while Multinomial Naïve Bayes performed efficiently but did not surpass SVM Linear in terms of accuracy. These findings suggest that SVM with Linear kernel is the most suitable model for this task.

## 5. Conclusion

Based on the results and discussion, it can be concluded that the detection of online gambling websites using text classification methods with Support Vector Machine (SVM) and Multinomial Naïve Bayes (MNB) was successfully carried out. A dataset of 1,225 URLs was used, comprising 769 gambling sites and 456 non-gambling sites. These URLs were processed using TF-IDF feature extraction and balanced using SMOTE to enhance classification performance. The results show that SVM with the RBF kernel achieved the best performance, with an accuracy of 91.88% and an F1-score of 93.70%, while other kernels and MNB produced lower results. The application of whitelist filtering also helped reduce false positives on trusted domains. These findings demonstrate that the combination of SVM and TF-IDF provides a reliable solution for detecting online gambling websites. However, future research can be directed toward expanding the dataset and applying semantic-based features or deep learning models to further enhance detection accuracy and generalization.

**Table 2.** Performance Comparison of Classification Models

| Model | Accuracy | Precision | Recall | Specificity | F1-Score |
|---|---|---|---|---|---|
| SVM Kernel Linier | 91,45% | **92,21%** | 94,67% | **85,71%** | **93,42%** |
| SVM Kernel RBF | **91,88%** | 91,72% | 96,00% | 84,52% | 93,70% |
| SVM Kernel Polynomial | 88,03% | 86,31% | **96,67%** | 72,62% | 91,20% |
| SVM Kernel Sigmoid | 88,03% | 86,31% | **96,67%** | 72,62% | 91,20% |
| SVM Multinomial Naïve Bayes | 88,46% | 90,20% | 92,00% | 82,14% | 91,00% |

## Acknowledgement
The author would like to thank the Gunadarma Foundation for supporting facilities during research.

## References

[1] M. R. Ardiansyah, K. Sudarmanto, K. Sukarna, and Z. Arifin, "Efektivitas Pemberantasan Tindak Pidana Judi Online," *Journal Juridisch*, vol. 1, no. 3, pp. 183–191, Dec. 2023, doi: 10.26623/JJ.V1I3.7946.

[2] N. F. Juhara, M. Amalia, and A. Mulyana, "Efektivitas Penegakan Hukum terhadap Judi Online di Indonesia: Analisis Yuridis dan Sosiologis," *Journal of Contemporary Law Studies*, vol. 2, no. 2, pp. 153–164, Feb. 2025, doi: 10.47134/LAWSTUDIES.V2I2.3353.

[3] N. Javier, B. D. Satoto, and Y. D. P. Negara, "Implementasi Teknik Web Scraping Untuk Pengumpulan Data Laporan Keuangan Perusahaan di Bursa Efek Indonesia (IDX)," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 2, pp. 2789–2795, Mar. 2025, doi: 10.36040/JATI.V9I2.13070.

[4] S. Hadijah, K. Auliasari, and F. X. Ariwibisono, "Peramalan Harga Saham Menggunalan Metode Simple Moving Average dan Web Scraping," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 2, pp. 1271–1278, Apr. 2024, doi: 10.36040/JATI.V8I2.9090.

[5] T. T. Widowati and M. Sadikin, "Analisis Sentimen Twitter terhadap Tokoh Publik dengan Algoritma Naive Bayes dan Support Vector Machine," *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, vol. 11, no. 2, pp. 626–636, Oct. 2020, doi: 10.24176/SIMET.V11I2.4568.

[6] M. N. Muttaqin and I. Kharisudin, "Analisis sentimen aplikasi gojek menggunakan support vector machine dan k nearest neighbor," *Unnes Journal of Mathematics*, vol. 10, no. 2, pp. 22–27, Dec. 2021, doi: 10.15294/UJM.V10I2.48474.

[7] A. Nurkholis, D. Alita, and A. Munandar, "Comparison of Kernel Support Vector Machine Multi-Class in PPKM Sentiment Analysis on Twitter," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 2, pp. 227–233, Apr. 2022, doi: 10.29207/RESTI.V6I2.3906.

[8] S. Rabbani, D. Safitri, N. Rahmadhani, A. A. F. Sani, and M. K. Anam, "Perbandingan Evaluasi Kernel SVM untuk Klasifikasi Sentimen dalam Analisis Kenaikan Harga BBM," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 3, no. 2, pp. 153–160, Oct. 2023, doi: 10.57152/malcom.v3i2.897.

[9] Cosmas Haryawan and Yosef Muria Kusuma Ardhana, "Analisa Perbandingan Teknik Oversampling SMOTE Pada Data Imbalanced," *Jurnal Informatika dan Rekayasa Elektronik*, vol. 6, no. 1, pp. 73–78, Apr. 2023, doi: 10.36595/jire.v6i1.834.

[10] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 91–99, Jun. 2022, doi: 10.1016/J.GLTP.2022.04.020.

[11] F. Rumaisa, Y. Puspitarani, A. Rosita, A. Zakiah, and S. Violina, "Penerapan Natural Language Processing (NLP) di bidang pendidikan," *Jurnal Inovasi Masyarakat*, vol. 1, no. 3, pp. 232–235, Dec. 2021, doi: 10.33197/JIM.VOL1.ISS3.2021.799.

[12] N. Afifah, D. Stiawan, and A. Bardadi, "Perbandingan Kinerja Kernel RBF dan Linear pada Algoritma Support Vector Machine

(SVM) untuk Prediksi Serangan Ransomware Locker," *JSI: Jurnal Sistem Informasi (E-Journal)*, vol. 15, no. 1, pp. 3118–3123, Apr. 2023, doi: 10.18495/jsi.v15i1.113.
[13] R. Chandra and E. M. Sipayung, "Analisis Sentimen Ulasan Aplikasi Samsat Digital Nasional Menggunakan Algoritma Naive Bayes Classifier," *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 10, no. 3, pp. 156–164, Jan. 2024, doi: 10.25077/TEKNOSI.V10I3.2024.156-164.
[14] G. Wijaya, D. Irawan, Z. Arifin, H. Oktavianto, M. Rahman, and G. Abdurrahman, "Studi Klasifikasi Topik Berita Dengan Algoritma Machine Learning," *J-ENSITEC*, vol. 11, no. 01, pp. 10202–10206, Dec. 2024, doi: 10.31949/JENSITEC.V11I01.12037.
[15] D. Septiani and I. Isabela, "SINTESIA: Analisa Term Frequency Inverse Document Frequency (TF-IDF) Dalama Temu Kembali Informasi Pada Dokumen Teks," Jurnal Sistem dan Teknologi Informasi Indonesia, 2022.
[16] A. R. Hanum *et al.*, "Analisis Kinerja Algoritma Klasifikasi Teks Bert dalam Mendeteksi Berita Hoaks," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 11, no. 3, pp. 537–546, Jul. 2024, doi: 10.25126/JTIIK.938093.
[17] M. Adam Rachman, E. Dyar Wahyuni Sistem Informasi, U. Pembangunan Nasional, J. Timur Jl Raya Rungkut Madya, and G. Anyar, "Komprasi Performa Model Klasifikasi Emosi Dengan Word Embedding Menggunakan Algoritma SVM dan Random Forest," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 2, pp. 2872–2878, Mar. 2025, doi: 10.36040/JATI.V9I2.13197.
[18] M. Afriansyah, J. Saputra, V. Yoga Pudya Ardhana, Y. Sa, and U. Qamarul Huda Badaruddin, "Algoritma Naïve Bayes Yang Efisien Untuk Klasifikasi Buah Pisang Raja Berdasarkan Fitur Warna," *Journal of Information Systems Management and Digital Business*, vol. 1, no. 2, pp. 236–248, Jan. 2024, doi: 10.59407/JISMDB.V1I2.438.
[19] S. A. Helmayanti, F. Hamami, and R. Y. Fa'rifah, "Penerapan Algoritma TF-IDF Dan Naive Bayes Untuk Analisis Sentimen Berbasis Aspek Ulasan Aolikasi Flip pada Google Play Store," *Jurnal Indonesia : Manajemen Informatika dan Komunikasi*, vol. 4, no. 3, pp. 1822–1834, Sep. 2023, doi: 10.35870/JIMIK.V4I3.415.
[20] J. T. Kumalasari and I. Puspitorini, "Perbandingan Metode Klasifikasi dan SMOTE Terhadap Analisa Sentimen Mobil Listrik Indonesia," *Jurnal Minfo Polgan*, vol. 13, no. 2, pp. 2257–2268, Jan. 2025, doi: 10.33395/jmp.v13i2.14428.
[21] I. Fadhli, Zulfaneti, and S. Junaidi, "Pengembangan Chatbot Informasi Seputar Nagari Berbasis Telegram Dengan Metode Natural Language Processing," *Ekasakti Engineering Journal* , vol. 4, no. 2, pp. 69–75, 2024, Accessed: Apr. 16, 2025. [Online]. Available: https://journal.unespadang.ac.id/EEJ/article/view/389