Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th June 2025. Vol. 4. No. 3; e-ISSN: 2808-4519

Improved Spam Email Detection Performance Based on Naïve Bayes Approach TF-IDF Vectorizer with Multi-Metric Optimization

Elpa Triana[1*], Ade Irma Purnamasari[2], Agus Bahtiar[3], Edi Tohidi[4]

[1,2,3,4] Teknik Informatika, STMIK IKMI Cirebon
Jl. Perjuangan No. 10B, Karyamulya, Kec.Kesambi, Kota Cirebon, Jawa Barat, Indonesia
elpatriana322@gmail.com[1*]

Abstract

Email spam has become a serious threat to user productivity and security in digital communication, particularly regarding malware and phishing risks. This study aims to develop and evaluate a more effective email spam detection system model using the Naïve Bayes algorithm optimized with TF-IDF Vectorizer, focusing on improving detection accuracy and handling language variations.The research methodology uses a Knowledge Discovery in Databases (KDD) approach with email message datasets collected from STMIK IKMI Cirebon students during the 2020-2024 period via Google Takeout. The data processing involves comprehensive preprocessing stages, including text cleaning, tokenization, stemming using Sastrawi for Indonesian, and data transformation using TF-IDF Vectorization. The model was evaluated using various data split ratios (90:10, 80:20, 70:30, and 60:40) to test system consistency and reliability. Experimental results show very satisfactory performance, with the 80:20 data split ratio achieving the highest accuracy of 92%. The model demonstrates a good balance between precision (0.94) for spam and (0.91) for non-spam, as well as recall values (0.91) for spam and (0.94) for non-spam. ROC Curve analysis yielded consistently high AUC values (0.96-0.97) across all data split ratios, indicating strong discriminative capability in distinguishing spam and legitimate emails. This research provides a significant contribution to developing more effective and efficient email filtering systems to protect users from various cyber threats.

.

Keywords: *Email spam detection, Naive Bayes, TF-IDF, Machine Learning, Email Filtering*

1. Introduction

Email has become an essential digital communication tool in daily life, both for personal and professional purposes. However, as email usage increases, security threats in the form of spam emails have become increasingly complex and dangerous. Spam emails not only disrupt user activities but also pose serious security risks, including the spread of malicious programs, fraud, potential financial losses, and personal data breaches.

According to recent digital security reports, approximately 55% of all circulating emails are spam, with more than 300 billion spam emails sent daily. In Indonesia, data from the National Cyber and Crypto Agency (BSSN) shows that email-based attacks accounted for 44.8% of total cyberattacks in 2023, with the majority being spam containing harmful content.
Several studies related to email spam detection and text classification have been conducted using various approaches. [1] developed an email spam filter using the Naïve Bayes algorithm connected to a mail server, demonstrating the ability to filter spam emails through text classification. [2] in their research "A review of spam email detection" analyzed the challenges of spam email detection, revealing that the dynamic digital environment and continuously evolving spammer strategies can cause performance degradation of detection systems with error rates reaching 48.81%. [3] in the study "A novel hybrid approach of SVM combined with NLP and probabilistic neural network for email phishing" developed a hybrid methodology for phishing detection by combining feature extraction, SVM classification, and Probabilistic Neural Network, which showed improved accuracy and precision compared to conventional methods. [4] in their research "Deep convolutional forest: a dynamic deep ensemble approach for spam detection in text" developed a dynamic ensemble model for spam detection using convolutional and pooling layers for feature extraction. The research achieved precision, recall, F1-score, and accuracy of 98.38%, proving the effectiveness of the dynamic ensemble approach for spam detection. [5] in the study "An Intelligent Framework Based on Deep Learning for SMS and e-mail Spam Detection" focused on the problem of low accuracy in small datasets using machine learning. This research applied multiple classifier machine learning and deep learning classifiers, with results showing that SVM provided the best performance among other classifiers.

Detecting spam emails is becoming increasingly complex as spam senders continue to develop new techniques to avoid filtering systems. The use of mixed languages, writing variations, and content that resembles legitimate emails are major challenges in developing effective spam detection systems.
The Naïve Bayes method has proven effective in text classification, including spam detection, due to its ability to process data quickly and accurately. To improve performance, this research uses TF-IDF Vectorizer, which allows higher weighting of keywords that distinguish

between spam and non-spam. The novelty of this research lies in [1] the combination of Naïve Bayes method and TF-IDF Vectorizer to improve spam detection accuracy, [2] special focus on spam emails in Indonesian campus environments, and [3] preprocessing of Indonesian-language emails using the Sastrawi method.

The main contributions of this research are [1] providing a new approach to email spam detection for the Indonesian context, [2] producing a more precise classification model, and [3] providing a reference for spam detection methodology in academic environments. This research has several limitations: [1] email data samples limited to the STMIK IKMI Cirebon environment, [2] using two main methods: Naïve Bayes and TF-IDF Vectorizer, and not including in-depth analysis of computational complexity. This research is important because it contributes to the development of more sophisticated email security systems, particularly for academic environments in Indonesia. The research results are expected to serve as a reference in developing more effective digital protection strategies.

2. Research Methodology

This research employs a quantitative approach using the Knowledge Discovery in Databases (KDD) method to develop an email spam detection system. The research design focuses on three main objectives: [1] developing a more accurate email spam detection system, [2] testing the effectiveness of combining Naïve Bayes and TF-IDF Vectorizer, and [3] analyzing the impact of preprocessing on spam detection accuracy. The methodology in this research consists of several stages, as shown in Figure 2.1 below:

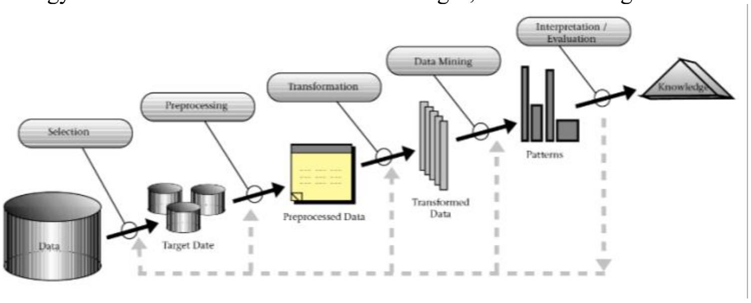

Figure 1: Research Methodology Stages

The following is an explanation of the stages shown in Figure 1:

1. Data Selection
   a. Data Collection
   Collecting email data from STMIK IKMI Cirebon students who are willing to provide their email data through Google Takeout. This data contains metadata information (such as sender, time) and message content from sent or received emails
   b. MBOX Extraction
   Extracting data from MBOX files obtained from Google Takeout to extract email content in a format that can be further analyzed.
2. Data Processing
   a. Stemmer Initialization Activating tools for stemming Indonesian words using the Sastrawi library. This stemming aims to return words to their basic form for easier analysis.
   b. Regex Pattern Compilation
   Creating data cleaning patterns using regex, such as removing HTML tags, URLs, and excessive spaces, to ensure email content is neater and easier to process.
   c. Text Cleaning
   Cleaning email content from irrelevant elements, such as HTML tags, URLs, excessive spaces, special characters, and common unimportant words (stop words) that could interfere with analysis results.
   d. Tokenization
   Breaking down the cleaned email content into small words (tokens) to facilitate further analysis.
   e. Spam Word Detection
   Using a list of common words in spam emails that have been simplified (stemmed) to detect potentially spam emails, such as emails containing words like 'promo,' 'cashback,' and 'free.'
   f. Data Upsampling
   Increasing the amount of data in the spam or non-spam category to balance the number of data points in both categories. This prevents the model from focusing only on the category with more data.
   g. Data Extraction to CSV
   Converting email data from MBOX format to CSV, which is easier to analyze. This CSV data is also labeled as 'spam' or 'non-spam' to aid the model training process.
3. Data Transformation
   Transforming email text into numbers using the TF-IDF Vectorizer technique, which considers word combinations (uni-grams and bi-grams). This technique makes it easier for machines to understand text mathematically.
4. Data Mining
   a. Using the Naïve Bayes algorithm to train the model to distinguish between spam and non-spam emails based on word patterns in email content.
   b. Varying the training data split with ratios of 90:10, 80:20, 70:30, and 60:40.
5. Evaluation
   a. Accuracy Score
   Measuring model accuracy by calculating the percentage of correct predictions from all tested data.
   b. Classification Report
   Creating a classification report showing metrics such as precision, recall, and F1-score to see how well the model recognizes emails as spam or non-spam.
   c. Confusion Matrix

Creating a confusion matrix, a table displaying correct and incorrect prediction results from the model, to determine how well the model correctly classifies emails.

d. ROC Curve

Generating an ROC curve to show the relationship between the True Positive Rate (rate of correct spam detection) and False Positive Rate. The Area

3. Result and Discussion

This research successfully developed an email spam detection system by integrating the Naïve Bayes Classifier method and TF-IDF Vectorizer. Through a series of stages in the Knowledge Discovery in Databases (KDD) process, the developed system demonstrated improved spam detection accuracy.

3.1. Classification Metric Evaluation

The model testing results showed very satisfactory performance. With an 80:20 training-test data split ratio, the model was able to achieve the highest accuracy of 92%. The precision value reached 0.94 for the spam category and 0.91 for the non-spam category. Meanwhile, the recall value reached 0.91 for spam and 0.94 for non-spam. The F1-score calculation yielded a value of 0.92, depicting a good balance between precision and recall.

Further analysis using the ROC curve produced consistently high Area Under Curve (AUC) values, ranging from 0.96 to 0.97 across various data split ratios. These results indicate strong discriminative ability of the model in distinguishing between spam and legitimate emails.

Table 1: Classification Performance Summary for 80:20 Ratio

Akurasi: 0.92					
		Precision	**Recall**	**F1- score**	**Support**
	Non-spam	0.91	0.94	0.92	2207
	Spam	0.94	0.91	0.92	2207
	accuracy			0.92	4414
	macro avg	0.92	0.92	0.92	4414
	weighted avg	0.92	0.92	0.92	4414

a. Accuracy: 0.92

An accuracy of 0.92 (92%) indicates that the model successfully predicted 92% of the total tested data (4414 data points) correctly. The accuracy formula is:

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Akurasi = \frac{2076 + 1999}{2076 + 131 + 208 + 1999}$$

$$Akurasi = \frac{4075}{4414}$$

$$Akurasi \approx 0.9232 \approx 0.92 = 92\%$$

From the total 4414 data points, most predictions correctly matched their original labels (non-spam and spam).

b. Precision measures the model's exactness - how many predictions are correct for data predicted to belong to a certain category. The formula is:

$$Precision\ (non - spam) = \frac{TN}{TN + FN}$$

$$Precision\ (non - spam) = \frac{2076}{2076 + 208} = \frac{2076}{2284} \approx 0{,}9090 \approx 91\%$$

$$Precision\ (spam) = \frac{TP}{TP + FP}$$

$$Precision\ (spam) = \frac{1999}{1999 + 131} = \frac{1999}{2130} \approx 0{,}9385 \approx 94\%$$

a) For the non-spam category, a precision of 0.91 shows that of all data predicted as non-spam, 91% were actually non-spam.

b) For the spam category, a precision of 0.94 shows that of all data predicted as spam, 94% were actually spam.

c. Recall measures the model's sensitivity, which is its ability to capture all actual data in a category. The formula is:

$$Recall\ (non-spam)\ = \frac{TN}{TN\ + FP}$$

$$Recall\ (non-spam)\ = \frac{2076}{2076+131} = \frac{2076}{2207} \approx 0{,}9406 \approx 94\%$$

$$Recall\ (spam)\ = \frac{TP}{TP\ + FN}$$

$$Recall\ (spam)\ = \frac{1999}{1999+208} = \frac{1999}{2207} \approx 0{,}9058 \approx 91\%$$

a) The recall for non-spam is 0.94, meaning the model successfully captured 94% of all data that actually belonged to the non-spam category.
b) The recall for spam is 0.91, meaning the model captured 91% of all data that actually belonged to the spam category.

d. F1-Score is the harmonic mean between precision and recall. The formula is:

$$F1-Score\ = 2\ x\ \frac{Precision \times Recall}{Precision + Recall}$$

$$F1(non-spam)\ = 2\ x\ \frac{0{,}91 \times 0{,}94}{0{,}91+0{,}94} = 2 \times \frac{0{,}8554}{1{,}85} = \frac{1{,}7100}{1{,}85} \approx 0{,}9248 \approx 92\%$$

$$F1(spam)\ = 2\ x\ \frac{0{,}94 \times 0{,}91}{0{,}94+0{,}91} = 2 \times \frac{0{,}8554}{1{,}85} = \frac{1{,}7106}{1{,}85} \approx 0{,}9248 \approx 92\%$$

The F1-Score for both non-spam and spam categories is 0.92, indicating a balance between precision and recall in classifying both categories.

e. Support
Support represents the number of actual data points in each category:
a) Non-spam: There are 2207 data points that actually belong to the non-spam category.
b) Spam: There are 2207 data points that actually belong to the spam category.
The total support is 4414, which is the total number of tested data points.

f. Macro Average and Weighted Average
a) Macro Average: This is the average precision, recall, and F1-Score for both categories without considering the number of data points per category. Its value is 0.92, reflecting balanced performance for both non-spam and spam categories.
b) Weighted Average: This is the average precision, recall, and F1-Score weighted based on the number of data points (support) for each category. Its value is also 0.92, showing that the balanced proportion of non-spam and spam data does not significantly affect the evaluation results.

3.2. Confusion Matrix

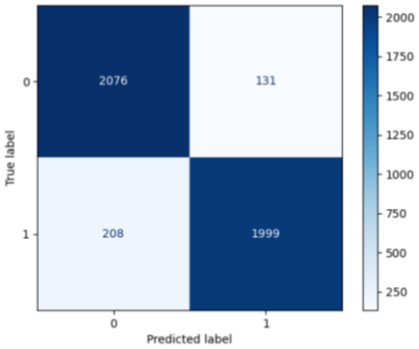

Figure 2: Visualization of Confusion Matrix for 80:20 Ratio

In Figure 2 above, this matrix shows the distribution of the model's predictions against the actual labels, with the following values:

a. True Negative (TN): A total of 2076 samples with original label 0 were correctly predicted as 0.
b. False Positive (FP): A total of 131 samples with original label 0 were incorrectly predicted as 1.
c. False Negative (FN): A total of 208 samples with original label 1 were incorrectly predicted as 0.
d. True Positive (TP): A total of 1999 samples with original label 1 were correctly predicted as 1.

The horizontal axis represents the predicted labels, while the vertical axis represents the actual labels. The colors in the matrix indicate the intensity of the number of samples, where darker colors indicate higher numbers.

From this matrix, the model demonstrates fairly good performance, as the majority of its predictions are accurate (2076 TN and 1999 TP). However, there are still some errors (131 FP and 208 FN) that need to be addressed to further improve accuracy.

3.4. ROC Curve

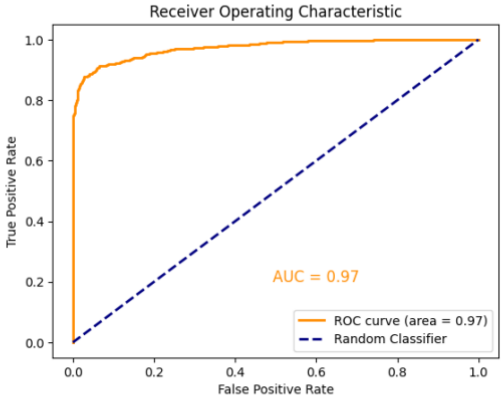

Figure 3: Visualization of ROC Curve for 80:20 Ratio

In Figure 3 above, the ROC graph shows the relationship between True Positive Rate (TPR) (vertical axis) and False Positive Rate (FPR) (horizontal axis) for various prediction thresholds.

a. True Positive Rate (TPR) or sensitivity is the ratio of correct positive predictions to the total number of positive labels.

b. False Positive Rate (FPR) is the ratio of incorrect positive predictions to the total number of negative labels.

The orange curve is the model's ROC curve, and the closer it gets to the upper left corner, the better the model's performance. The AUC (Area Under the Curve), which has a value of 0.97, indicates that the model has excellent performance in distinguishing between positive and negative classes (with a maximum score of 1.0).

The blue dashed line is the baseline for a random model, which shows performance without classification ability (AUC = 0.5). In this case, an AUC of 0.97 signifies that the model is much better than random guessing.

3.5. Efektivitas Integrasi Naïve Bayes dan TF-IDF

The combination of the Naïve Bayes Classifier algorithm and TF-IDF Vectorizer technique has proven effective in improving email spam detection accuracy. Naïve Bayes is capable of classifying emails well based on the probability of word occurrences, while the TF-IDF Vectorizer plays a role in converting email text into more informative numerical representations.

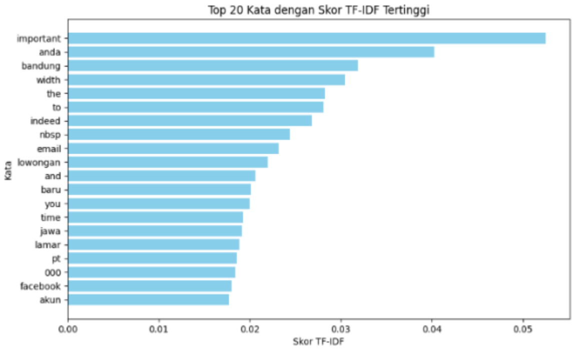

Figure 4: Visualization of Plot top words tf-idf

4. Conclusion

Based on the research conducted, the development of an email spam detection system using a combination of the Naïve Bayes Classifier algorithm and TF-IDF Vectorizer technique successfully achieved a high accuracy rate of 92% at an 80:20 training-test data split ratio, fulfilling the objective of developing a more accurate detection system. Testing the effectiveness of the combination of these two methods showed significant results, where the integration of Naïve Bayes and TF-IDF consistently outperformed the use of single methods in capturing complex spam patterns, as evidenced by high Area Under Curve (AUC) values between 0.96 and 0.97 and low classification error rates in the confusion matrix. Analysis of the influence of data preprocessing such as text cleaning, stemming, tokenization, and stopwords removal demonstrated substantial contributions in enhancing the model's ability to identify spam indicators more accurately, resulting in an effective email filtering system to protect users from spam threats. For further development, it is suggested to: (1) explore deep learning methods to improve detection accuracy, (2) implement adaptive learning to accommodate evolving spam patterns, (3) add visual content analysis in the detection process, and (4) optimize real-time performance by considering computational efficiency.

References

[1] H. Mukhtar, J. Al Amien, and M. A. Rucyat, “Filtering Spam Email menggunakan Algoritma Naïve Bayes,” *Jurnal CoSciTech (Computer Science and Information Technology)*, vol. 3, no. 1, pp. 9–19, May 2022, doi: 10.37859/coscitech.v3i1.3652.

[2] F. Jáñez-Martino, R. Alaiz-Rodríguez, V. González-Castro, E. Fidalgo, and E. Alegre, “A review of spam email detection: analysis of spammer strategies and the dataset shift problem,” *Artif Intell Rev*, vol. 56, no. 2, pp. 1145–1173, Feb. 2023, doi: 10.1007/s10462-022-10195-4.

[3] A. Kumar, J. M. Chatterjee, and V. G. Díaz, "A novel hybrid approach of SVM combined with NLP and probabilistic neural network for email phishing," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 1, pp. 486–493, 2020, doi: 10.11591/ijece.v10i1.pp486-493.

[4] M. A. Shaaban, Y. F. Hassan, and S. K. Guirguis, "Deep convolutional forest: a dynamic deep ensemble approach for spam detection in text," *Complex and Intelligent Systems*, vol. 8, no. 6, pp. 4897–4909, Dec. 2022, doi: 10.1007/s40747-022-00741-6.

[5] U. Maqsood, S. Ur Rehman, T. Ali, K. Mahmood, T. Alsaedi, and M. Kundi, "An Intelligent Framework Based on Deep Learning for SMS and e-mail Spam Detection," *Applied Computational Intelligence and Soft Computing*, vol. 2023, 2023, doi: 10.1155/2023/6648970.

[6] N. Ahmad, S. Hafizh, and R. Sulthanah, "Prediksi Kelulusan Mata Kuliah Mahasiswa Teknologi Informasi Menggunakan Algoritma K-Nearest Neighbor The Prediction for Graduation for Information Technology Student ' s Course Using The K-Nearest Neighbor Algorithm," vol. 14, pp. 135–149, 2024.

[7] M. Gratia, B. Sitorus, N. Maria, and Y. N. Safa, "Tinjauan Literatur Manajemen Risiko Cyber dalam Proyek : Identifikasi , Evaluasi , dan Mitigasi Ancaman Literature Review Cyber Risk Management in Projects : Threat Identification , Evaluation and Mitigation," vol. 14, pp. 187–198, 2024.

[8] N. N. Sari, T. T. Anisah, and R. Fitriani, "Implementasi Machine Learning untuk Prediksi Harga Laptop Menggunakan Algoritma Regresi Linear Berganda Machine Learning Implementation for Laptop Price Prediction Using Multiple Linear Regression Algorithm," vol. 14, pp. 162–177, 2024.

[9] S. A. Brown, B. A. Weyori, A. F. Adekoya, P. K. Kudjo, and S. Mensah, "Predicting Blocking Bugs with Machine Learning Techniques : A Systematic Review," vol. 13, no. 6, pp. 674–683, 2022.

[10] S. Senhadji, R. Azad, and S. Ahmed, "Fake News Detection Using Naïve Bayes and Long Short Term Memory Fake news detection using naïve Bayes and long short term memory algorithms," no. March, pp. 746–752, 2022, doi: 10.11591/ijai.v11.i2.pp746-752.

[11] F. J. Martino, R. A. Rodríguez, and V. G. Castro, "A review of spam email detection : analysis of spammer strategies and the dataset shift problem," *Artif. Intell. Rev.*, vol. 56, no. 2, pp. 1145–1173, 2023, doi: 10.1007/s10462-022-10195-4.

[12] K. S. Putri, I. R. Setiawan, A. Pambudi, A. Sentimen, and N. B. Classifier, "'Technologia' Vol 14, No. 3, Juli 2023 227 ANALISIS SENTIMEN TERHADAP BRAND SKINCARE LOKAL MENGGUNAKAN NAÏVE BAYES CLASSIFIER," vol. 14, no. 3, pp. 227–232, 2023.

[13] J. Al Amien, H. Mukhtar, and M. A. Rucyat, "Jurnal Computer Science and Information Technology (CoSciTech)," vol. 3, no. 1, pp. 9–19, 2022.

[14] N. Agustina and M. Hermawati, "Implementasi Algoritma Naïve Bayes Classifier untuk Mendeteksi Berita Palsu pada Sosial Media," vol. 14, no. 4, pp. 206–213, 2021, doi: 10.30998/faktorexacta.v14i4.11259.

[15] J. S. Komputer, "Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter," vol. 5, no. September, pp. 697–711, 2021.

[16] E. Gbenga, J. Stephen, H. Chiroma, A. Olusola, and O. Emmanuel, "Heliyon Machine learning for email spam fi ltering : review , approaches and open research problems," vol. 5, no. February, 2019, doi: 10.1016/j.heliyon.2019.e01802.

[17] M. R. Qisthiano, T. B. Kurniawan, E. S. Negara, and M. Akbar, "Pengembangan Model Untuk Prediksi Tingkat Kelulusan Mahasiswa Tepat Waktu dengan Metode Naïve Bayes," vol. 5, pp. 987–994, 2021, doi: 10.30865/mib.v5i3.3030.

[18] C. Herdian, M. Quinn, and S. Margareta, "Perbandingan Algoritma Naive Bayes di dalam Scikit-Learn Python Library dengan Murni Algoritma Naive Bayes : Studi Kasus Klasifikasi Email Berbahaya," vol. 9, no. 1, pp. 1–10, 2024.

[19] R. Sistem, J. W. Iskandar, Y. Nataliani, F. T. Informasi, U. Kristen, and S. Wacana, "JURNAL RESTI," vol. 5, no. 158, pp. 1120–1126, 2021.

[20] Y. F. Hassan and S. K. Guirguis, "Deep convolutional forest : a dynamic deep ensemble approach for spam detection in text," *Complex Intell. Syst.*, vol. 8, no. 6, pp. 4897–4909, 2022, doi: 10.1007/s40747-022-00741-6.

[21] M. F. Madjid, D. E. Ratnawati, and B. Rahayudi, "Sentiment Analysis on App Reviews Using Support Vector Machine and Naïve Bayes Classification," vol. 7, no. 1, pp. 556–562, 2023.

[22] R. Blanquero, E. Carrizosa, and P. Ramírez-cobo, "Computers and Operations Research Variable selection for Naïve Bayes classification," vol. 135, 2021.

[23] D. Fitria, Y. Cahyana, D. Sulistya, and K. A. Baihaqi, "Pemilihan Algoritma Terbaik Untuk Klasifikasi Jenis E-Mail dengan Metode TF-IDF," *J. Ris. Sist. Inf. Dan Tek. Inform.*, vol. 9, no. 1, pp. 398–407, 2024, [Online]. Available: https://tunasbangsa.ac.id/ejurnal/index.php/jurasik