Application of Text Classification for Sentiment Analysis on E-Commerce Product Reviews

Adithia Gunawan[1], Adriel Walintukan[2], Julio Weol[3], Mutiara Makarawung[4], Cleymency Kaparang[5]

[1,2,3,4,5]Universitas Klabat; Jl. Arnold Mononutu, Airmadidi Bawah, Kec. Airmadidi, Kabupaten Minahasa Utara, Sulawesi Utara, 95371

[1,2,3,4,5]Ilmu Komputer/Sistem Informasi, Fakultas Ilmu Komputer, Universitas Klabat, Airmadidi

e-mail: [1]s22310014@student.unklab.ac.id, [2]s22310041@student.unklab.ac.id,[3]s22219020@student.unklab.ac.id, [4]s22310105@student.unklab.ac.id, [5]s22210615@student.unklab.ac.id

Abstract

This study aims to perform sentiment classification on e-commerce product reviews using the Naive Bayes algorithm. The dataset used is PRDECT-ID, consisting of 5,400 customer review entries. The research process involves data preprocessing steps including case transformation, tokenization, stopword removal, and stemming, conducted using the Process Documents from Data operator in RapidMiner. The model was trained and validated using the Cross Validation method and evaluated using accuracy, precision, recall, and F1-score metrics derived from the confusion matrix. The evaluation results show that the Naive Bayes model performs exceptionally well, achieving an accuracy of 99.52%, precision of 99.77%, recall of 99.22%, and F1-score of 99.49%. These results indicate that Naive Bayes is an effective, efficient, and reliable algorithm for sentiment analysis on product reviews in e-commerce platforms.

Keywords— Sentiment Analysis, E-commerce, Product Reviews, Naive Bayes, Text Classification

1. INTRODUCTION

The rapid development of digital technology has brought major changes in various aspects of life, including in the field of commerce. One of the biggest transformations that has occurred is the emergence of electronic commerce (e-commerce), where the process of buying and selling goods or services is done online through a digital platform. This phenomenon has significantly changed people's consumption patterns because it provides convenience in transactions anytime and anywhere. Indonesia itself is one of the countries with the fastest e-commerce growth in the world, and platforms such as Shopee and Tokopedia are the main choices for people to do online shopping activities [1].

In the context of e-commerce, product reviews provided by consumers play an important role. These reviews are not only a reference for potential buyers to assess the quality of a product or service, but also a very valuable source of data for businesses. Through reviews, companies can understand customer needs, satisfaction, and complaints about the products they offer. However, the huge volume of unstructured review data makes it difficult to analyze manually. Therefore, a technology-based solution is needed that is able to analyze consumer opinions automatically and efficiently [5]. One effective solution is sentiment analysis, which is a process of classifying text based on its emotional orientation into positive, negative, or neutral categories [9].

Sentiment analysis is part of *Natural Language Processing* (NLP) techniques that allow systems to understand and extract opinions from large amounts of text [10]. This technique is widely applied to product reviews, social media, and online discussion forums as an aid in strategic business decision-making [12], [17], [18]. According to Pang et al., sentiment classification is more challenging than traditional topic-based text classification because it involves interpreting emotions and subjective opinions from text data [10]. The use of machine learning algorithms in sentiment analysis enables automatic, scalable, and efficient interpretation of consumer opinions, especially with the increasing volume of e-commerce reviews [3], [10], [20].

In this research, the main focus is the application of the *Text Classification* method for sentiment analysis on e-commerce product reviews using the Naive Bayes algorithm. Naive Bayes is a probabilistic-based classification algorithm that is widely used in text classification tasks because it is simple, efficient, and has competitive performance compared to other algorithms in the context of high-dimensional text data [2]. Naive Bayes is also effective in handling large datasets and is able to provide fairly accurate results even with the assumption of independence between features [2], [13], [16]. Research by Zhang [13] demonstrated the optimality of Naive Bayes even under strong independence assumptions. Furthermore, McCallum and Nigam [14] found that the multinomial model of Naive Bayes is more effective for text classification tasks compared to the Bernoulli model.

Various previous studies have proven the effectiveness of Naive Bayes in sentiment analysis tasks. Research conducted by Muzaki et al. [2] showed that the Naive Bayes method was successfully developed in the form of a web-based system capable of performing sentiment analysis automatically on product reviews on e-commerce platforms. The system proved to be efficient in classifying reviews into positive and negative sentiments and provided convenience in managing consumer opinions by businesses.

Meanwhile, research by Pradana and Wibowo [3] applied the Multinomial Naive Bayes algorithm to analyze local product reviews (Compass Shoes) on Tokopedia. This research not only succeeded in grouping reviews into two sentiment categories, but also displayed keyword visualization through wordcloud, which can be used by companies to design product improvement strategies based on customer perceptions. The results show a high level of accuracy, reaching 92.31%, thus proving the reliability of this method in the context of e-commerce.

Adiguna and Pramudya's research [4] compared the performance of Naive Bayes with other algorithms such as Support Vector Machine (SVM) and Random Forest in analyzing Shopee app reviews. The results showed that Naive Bayes obtained an accuracy of 84.76%, indicating that although it is not the algorithm with the highest accuracy, NB is still worthy of being used as an alternative because of its efficiency in model training and result interpretation.

Apart from the local context, an international study conducted by Haroon et al. [7] also confirmed the superiority of Naive Bayes over other algorithms such as Logistic Regression and SVM in analyzing customer review sentiment. In the study, Naive Bayes obtained an accuracy of 94%, higher than other algorithms in the same experiment. This shows the potential of Naive Bayes to be widely used in various opinion analysis contexts.

The study of Rizqullah et al. [5] also supports the importance of automated analysis of user reviews in e-commerce platforms. Despite using the SVM algorithm, the study emphasized that text classification is helpful in identifying aspects that need to be improved from e-commerce services. These findings suggest that machine learning algorithm-based text analysis approaches are highly relevant in addressing the needs of today's digital businesses.

Based on this background , this research aims to apply the text classification method using the Naive Bayes algorithm in conducting sentiment analysis of e-commerce product reviews. With this approach, it is expected that the research can make a real contribution in developing a system that is able to help businesses understand consumer sentiment more accurately, quickly, and efficiently, so that business decisions can be made more targeted.

2. RESEARCH METHODS

2.1. Data Preparation

The dataset used is the PRDECT-ID Dataset which is obtained from the Kaggle platform https://www.kaggle.com/datasets/jocelyndumlao/prdect-id-indonesian-emotion-classification. This dataset contains data about e-commerce products along with customer reviews. The main focus is on sentiment analysis. The columns in this data include Category, Product Name, Location, Price, Overall Rating, Number Sold, Total Reviews, Customer Rating, Customer Review, Sentiment, Emotion. This data set consists of 5400 data. According to Dey et al. [8], sentiment analysis on large datasets is crucial to capture diverse opinions and detect sentiment variations more accurately.

	A	B	C	D	H	I	J	K
1	Category	Product Name	Location	Price	Customer	Customer Review	Sentiment	Emotion
2	Computers	Wireless Keybo	Jakarta Ut	53500	5	Alhamdulillah berfungs	Positive	Happy
3	Computers	PAKET LISENSI	Kota Tang	72000	5	barang bagus dan resp	Positive	Happy
4	Computers	SSD Midasforce	Jakarta Ba	213000	5	barang bagus, berfungs	Positive	Happy
5	Computers	ADAPTOR CHAI	Jakarta Tir	55000	5	bagus sesuai harapan p	Positive	Happy
6	Computers	ADAPTOR CHAI	Jakarta Tir	55000	5	Barang Bagus, pengem	Positive	Happy
					5	Barusan saya tes untuk	Positive	Happy
5395	Household	PCK-01 Penjepi	Jakarta Pu	35000	5	Barangnya sudah samp	Positive	Happy
5396	Household	PCK-01 Penjepi	Jakarta Pu	35000	5	produk sesuai,, bergun	Positive	Happy
5397	Household	PCK-01 Penjepi	Jakarta Pu	35000	5	Harga bersaing, barang	Positive	Love
5398	Household	Ultrasonic Aror	Jakarta Ut	99000	5	Beli ini krn Anak & Istri	Positive	Love
5399	Household	Ultrasonic Aror	Jakarta Ut	99000	5	pengemasan barang ba	Positive	Happy
5400	Household	Ultrasonic Aror	Jakarta Ut	99000	5	Mungil tapi bekerja dng	Positive	Happy
5401	Household	TDS Meter 3 Al	Jakarta Ut	14400	5	Produk sesuai deskripsi	Positive	Love

Figure 1. PRDECT-ID Dataset

2.2. Data Preprocessing

Before the modeling process is carried out, the data cleaning stage is first carried out on the dataset to be used. This process aims to ensure that the data is clean, complete, and consistent, so as to improve the accuracy of the modeling results. In this data cleaning stage, the Filter Example operator is used which is set with the class: not missing label condition. This setting aims to filter the data so that only examples (records) that have complete class labels will be processed further, while data that has empty or missing labels will be ignored. With this approach, it is expected that the resulting model will be able to provide more accurate and reliable predictions or analysis.

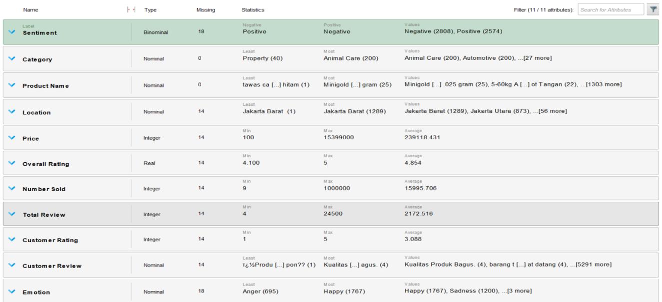

Figure 2. Before Data Cleaning

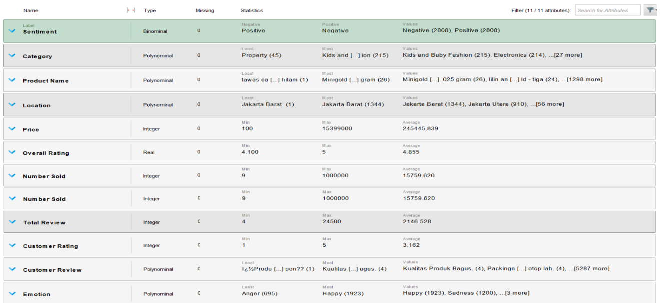

Figure 3. After Data Cleaning

2.3. *Data Modeling*

In the Data Modeling stage, before the data is fed into the classification algorithm, text processing is first performed using the Process Documents from Data operator. This operator is used to convert raw text data into numerical representations that can be processed by machine learning algorithms. Within this operator there are a series of text preprocessing stages, namely:

- Transform Cases: At this stage, all letterforms are converted to lowercase to avoid redundancy in word processing [8]. This transformation is important to ensure uniformity, as case sensitivity can affect word matching during analysis.
- Tokenize: This is a text processing stage that breaks down sentences into individual word units to improve interpretability. This process aims to produce word segmentation which will then be classified as entities and calculated document metric values for further analysis [6].
- Filter Stopwords: Stopwords are common words (such as "the", "is", "at") that do not contribute significant meaning to text analysis. Removing these words helps to reduce noise and improve computational efficiency [11]. In a study by Zhang [11], the removal of stopwords significantly improved the classification accuracy of sentiment analysis.
- Stem: It is an essential stage that transforms a compound word into a base word through the elimination of affixes, which include prefixes, infixes, suffixes, and confixes [6]. The accuracy of the stemming process has a significant impact on the validity of the analysis results, given its role as a fundamental component in text processing [4].

Once this process is complete, the result is a representation of the data in the form of a term vector that is ready to be used for modeling. Next, the cleaned and processed data is fed into the Cross Validation process to build and evaluate the model. The classification model used is Naive Bayes, which is trained using the preprocessed data. The training process is performed in the Training section, and the model results are then applied to the test data in the Testing section using the Apply Model operator. With thorough text preprocessing and systematically validated modeling, this process is expected to produce accurate and reliable classification models.

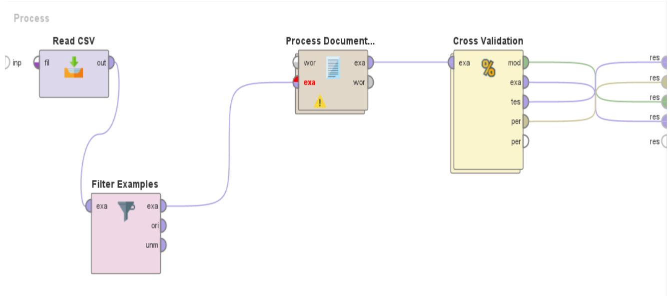

Figure 4. Operators Used

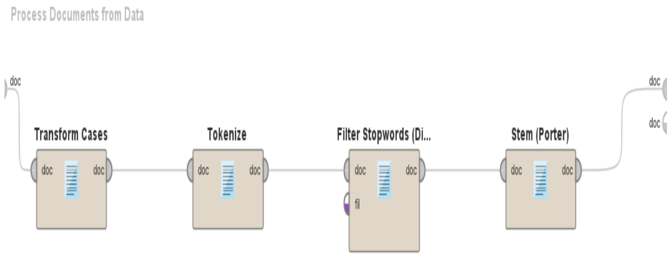

Figure 5. Content Operators Process Document From Data

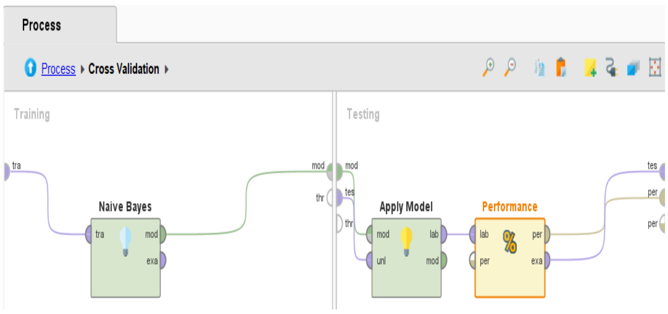

Figure 6. Model Training and Evaluation

2.4. Evaluation

To evaluate the performance of the built model, several metrics are used: Accuracy, Precision, Recall, and F1-Score. These metrics are derived from the *Confusion Matrix*, which shows the number of correct and incorrect predictions for each class. Accuracy measures the proportion of correctly classified instances, Precision measures the correctness of positive predictions, Recall assesses the model's ability to identify all positive samples, and F1-Score represents the harmonic mean of Precision and Recall [14]. According to Caruana and Niculescu-Mizil [15], these metrics are crucial for understanding the strengths and weaknesses of the model, especially in unbalanced datasets.

Table 1. Confusion Matrix Table

Class	**Positive Classified**	**Negative Classified**
Positive	TP (True Positive)	FN (False Negative)

Negative	FP (False Positive)	TN (True Negative)

In measuring performance using a confusion matrix, there are 4 terms as a representation of the results of the classification process. The four terms are True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). The True Negative (TN) value is the amount of negative data detected correctly, while False Positive (FP) is negative data but detected as positive data. Meanwhile, True Positive (TP) is positive data that is detected correctly. False Negative (FN) is the opposite of True Positive, so it is positive data, but detected as negative data. Therefore, to support the evaluation, formulas for calculating each metric based on the values obtained from the Confusion Matrix will also be written.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

where:

- TP is True Positive, which is the amount of positive data that is correctly classified by the system.
- TN is True Negative, which is the amount of negative data that is correctly classified by the system.
- FN is False Negative, which is the amount of negative data but misclassified by the system.
- FP is False Positive, which is the amount of positive data but misclassified by the system.

3. RESULT AND DISCUSSION

3.1. Modeling Proses Text Classification

Table 2 shows the results of each stage in the text pre-processing process performed using the Process Documents from Data operator in RapidMiner. This stage starts from the initial input in the form of Customer Review which is still in the form of raw text. The first process is Transform Cases, which converts all letters into lowercase letters to avoid redundancy in word processing. Then, the text is broken into single words (tokens) using the Tokenize operator. The next stage is Filter Stopwords, which removes common words that do not make an important contribution to the analysis, such as conjunctions or prepositions. Finally, a Stemming process is carried out with the Porter algorithm to convert words to their basic form, for example the word “function” becomes “function” and “packaging” becomes the root form. This whole process is important to ensure that the text data is ready to be used in the classification modeling process with more optimal results.

Table 2. Result Process Document to Data

Process	Result
Customer Review	Barang Bagus, pengemasan Aman, dapat Berfungsi dengan Baik

Transform Cases	barang bagus, pengemasan aman, dapat berfungsi dengan baik
Tokenize	“barang”, “bagus”, “pengemasan”, “aman”, “dapat”, “berfungsi”, “dengan”, “baik”
Filter Stopwords	“barang”, “bagus”, “pengemasan”, “aman”, “berfungsi”
Stem	“barang”, “bagus”, “kemas”, “aman”, “fungsi”

3.2. Evaluation

The evaluation is carried out using the confusion matrix method which aims to measure the performance of the classification model in classifying product review data into positive or negative sentiment categories. In this research, the evaluation metrics used include accuracy, precision, recall, and F1-score. These metrics provide an overall picture of how well the model classifies the data correctly. Accuracy indicates the proportion of correct classifications out of all data, precision measures the accuracy of predicting positive classes, recall describes the model's ability to find all correct positive data, while F1-score is a harmonization between precision and recall.

Table 3. Result Confusion Matrix Table

Class	true Positive	true Negative
pred. Positive	2554	6
pred. Negative	20	2802

- $$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$
  $$Accuracy = \frac{2554 + 2802}{2554 + 2802 + 6 + 20}$$
  $$Accuracy = \frac{5356}{5382}$$
  $$Accuracy = 99.52\%$$

- $$Precision = \frac{TP}{TP + FP}$$
  $$Precision = \frac{2554}{2554 + 6}$$
  $$Precision = \frac{2554}{2560}$$
  $$Precision = 99.77\%$$

- $$Recall = \frac{TP}{TP + FN}$$
  $$Recall = \frac{2554}{2554 + 20}$$
  $$Recall = \frac{2554}{2574}$$
  $$Recall = 99.22\%$$

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

  $F1 - Score = 2 \times \frac{0.9977 \times 0.9922}{0.9977 + 0.9922}$

  $F1 - Score = 2 \times \frac{0.9899}{1.9899}$

  $F1 - Score = 99.49\%$

Table 4 Naive Bayes Model Performance Evaluation Table

Metrics	Result
Accuracy	99.52%
Precision	99.77%
Recall	99.22%
F1-Score	99.49%

Based on the evaluation results shown in Table 3, it can be concluded that the classification model used is able to provide excellent performance, indicated by high values of accuracy, precision, recall, and F1-score.

4. CONCLUSION

Based on the results of the research, the application of text preprocessing techniques such as *Transform Cases*, *Tokenization*, *Stopword Removal*, and *Stemming* has proven crucial in producing clean and consistent product review data. This significantly impacts the performance of the Naive Bayes algorithm in classifying product review sentiments. The developed model demonstrates very high performance, with an accuracy of 99.52%, precision of 99.77%, recall of 99.22%, and F1-score of 99.49%. These results indicate that the model is capable of accurately distinguishing between positive and negative reviews.

The effectiveness of Naive Bayes in sentiment analysis is supported by various studies. Zhang [11] showed that Naive Bayes remains optimal even under strong independence assumptions, making it highly efficient for large datasets. Furthermore, the *Multinomial Naive Bayes* model, as highlighted by McCallum and Nigam [8], is particularly effective for text classification due to its ability to leverage term frequencies. Studies by Caruana and Niculescu-Mizil [15] also confirm that Naive Bayes consistently achieves competitive performance against more complex algorithms like *Support Vector Machine* (SVM) and *Random Forest* in various sentiment analysis tasks.

In addition, the stability of Naive Bayes, even when the independence assumption is partially violated, makes it a robust choice for sentiment analysis on diverse and unstructured data, as shown by Rish [16]. This characteristic is particularly advantageous for e-commerce platforms where user reviews are abundant and vary greatly in format and language. Moreover, the study by Ng and Jordan [14] indicates that Naive Bayes converges faster to its asymptotic error compared to *Logistic Regression*, proving its suitability for high-dimensional text data.

Therefore, the implementation of Naive Bayes for sentiment analysis in e-commerce product reviews demonstrates a reliable, efficient, and scalable solution to interpret consumer opinions. This capability allows businesses to gain deeper insights into customer satisfaction, identify critical product issues, and make data-driven decisions that can enhance service quality and product development. The integration of automated sentiment analysis systems could greatly benefit e-commerce platforms in monitoring customer opinions in real-time and responding proactively to market needs.

REFERENCES

[1] A. Rafid Rizqullah, A. Wedhasmara, R. Izwan Heroza, A. Putra, and P. Putra, "ANALISIS MASALAH PADA DATA REVIEW APLIKASI TERHADAP LAYANAN E-COMMERCE MENGGUNAKAN METODE TEXT CLASSIFICATION." Jurnal TEKNO KOMPAK, vol. 16, no. 1, pp. 186–198, 2023.

[2] A. Muzaki *et al.*, "ANALISIS SENTIMEN PADA ULASAN PRODUK DI E-COMMERCE DENGAN METODE NAIVE BAYES," *Jurnal Riset dan Aplikasi Mahasiswa Informatika (JRAMI)*, vol. 05, 2024.

[3] D. A. Pradana and A. P. Wibowo, "ANALISIS SENTIMEN ULASAN PRODUK SEPATU COMPASS DI E-COMMERCE TOKOPEDIA MENGGUNAKAN ALGORITMA NAIVE BAYES CLASSIFIER (NBC)" Inovasi Pembangunan: Jurnal Kelitbangan, vol. 12, no. 3, pp. 1–4, 2024.

[4] V. Brilian Adiguna and R. A. Pramudya, "Analisis Sentimen Ulasan Aplikasi Shopee Menggunakan Algoritma Random Forest, Naïve Bayes, dan Support Vector Machine di Kota Semarang." Digital Business Intelligence Journal, vol. 1, no. 1, pp. 38–53, 2025.

[5] M. Haroon, Z. Alam, R. Kousar, J. Ahmad, and F. Nasim, "Sentiment Analysis of Customer Reviews on E-commerce Platforms: A Machine Learning Approach," *Bulletin of Business and Economics (BBE)*, vol. 13, no. 3, pp. 230–238, Aug. 2024

[6] I. P. Rahayu, A. Fauzi, and J. Indra, "Analisis Sentimen Terhadap Program Kampus Merdeka Menggunakan Naive Bayes Dan Support Vector Machine," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 4, no. 2, p. 296, Dec. 2022

[7] A. Alamsyah and F. Saviera, "A Comparison of Indonesia E-Commerce Sentiment Analysis for Marketing Intelligence Effort," *ArXiv Preprint*, vol. 2103.00231, 2021.

[8] L. Dey, S. Chakraborty, A. Biswas, B. Bose, and S. Tiwari, "Sentiment Analysis of Review Datasets Using Naive Bayes and K-NN Classifier," *ArXiv Preprint*, vol. 1610.09982, 2016.

[9] V. Narayanan, I. Arora, and A. Bhatia, "Fast and accurate sentiment classification using an enhanced Naive Bayes model," *ArXiv Preprint*, vol. 1305.6143, 2013.

[10] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment Classification using Machine Learning Techniques," *ArXiv Preprint*, vol. cs/0205070, 2002.

[11] H. Zhang, "The Optimality of Naive Bayes," *Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference (FLAIRS 2004)*, 2004.

[12] A. McCallum and K. Nigam, "A comparison of event models for Naive Bayes text classification," *AAAI Workshop on Learning for Text Categorization*, 1998.

[13] J. Rennie, L. Shih, J. Teevan, and D. Karger, "Tackling the poor assumptions of naive Bayes text classifiers," *Proceedings of the Twentieth International Conference on Machine Learning (ICML 2003)*, 2003.

[14] A. Y. Ng and M. I. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes," *Neural Information Processing Systems (NIPS 2002)*, 2002.

[15] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," *Proceedings of the 23rd International Conference on Machine Learning (ICML 2006)*, 2006.

[16] I. Rish, "An empirical study of the naive Bayes classifier," *Proceedings of the IJCAI Workshop on Empirical Methods in AI*, 2001.

[17] D. J. Hand and K. Yu, "Idiot's Bayes—not so stupid after all?," *The Statistician*, vol. 51, no. 1, pp. 69–76, 2001.

[18] J. Chen, Z. Dai, J. Duan, H. Matzinger, and I. Popescu, "Naive Bayes with Correlation Factor for Text Classification Problem," *ArXiv Preprint*, 2019.

[19] C. Kaya, Z. H. Kilimci, M. Uysal, and M. Kaya, "Migrating Birds Optimization-Based Feature Selection for Text Classification," *ArXiv Preprint*, 2024.

[20] H. J. Escalante, M. Montes-y-Gómez, L. Villaseñor-Pineda, and M. L. Errecalde, "Early text classification: a Naive solution," *ArXiv Preprint*, 2015.