U.Porto Journal of Engineering, 10:3 (2024) 23-38
ISSN 2183-6493
DOI: 10.24840/2183-6493_0010-003_002464

Received: 22 November, 2023
Accepted: 1 July, 2024
Published: 10 July, 2024

A Comparative Analysis of Resource-Efficient Machine Learning Models in News Categorization

Mohammad Hossein Zolfagharnasb
Department of Electrical and Computer Engineering, Faculty of Engineering, University of Porto, Rua Dr. Roberto Frias, 4200-465 PORTO, Portugal (up202300418@edu.fe.up.pt) ORCID 0000-0001-6124-7507

Siavash Damari
Department of Statistics, Mathematics, and Computer Science, University of Allameh Tabataba'i, Western Azadi Stadium Blvd, Tehran, Iran (siavashdamari77@gmail.com) ORCID 0009-0002-8486-9549

Author Keywords

Real-time Content Classification, News Categorization, Natural Language Processing (NLP), Machine Learning.

Type: Research Article

Open Access
Peer Reviewed

Abstract

The constant stream of news nowadays highlights the necessity for meticulous assessment to ensure that the information accurately reaches its intended audience with the least amount of delay least delay. Despite the flexibility and efficiency of Deep Learning (DL) models, their intricate training and substantial resource demands pose significant challenges for their deployment in real-time applications. In this regard, this study evaluates the performance of resource-efficient Machine Learning (ML) techniques – Multinomial Naive Bayes (MNB), Random Forest (RF), Support Vector Machine (SVM), and Logistic Regression (LR) – in categorizing news. Based on the results, all the evaluated models attain a commendable level of accuracy in news categorization. Notably, the SVM excels, achieving an accuracy rate of 98% and a mean squared error of 0.28. This performance exemplifies the robust effectiveness of classical ML models in the categorization of news, particularly when enhanced by a suitably tailored preprocessing pipeline.

1. Introduction

In the current digital landscape, the expansion of social media, blogs, and online news platforms has generated an enormous amount of news data daily (Husin 2023). Thus, the accurate evaluation of this information and extracting relevant and trustworthy data have become increasingly crucial (M. Zolfagharnasab et al. 2022). For this reason, news classification is a powerful tool in information and media that helps audiences quickly access the news they are interested in and enables media and news organizations to improve their programs (Truică and Apostol 2023). News classification can also provide valuable information for various analyses and effective decision-making. Other importance of news classification includes reducing redundant information and organizing data, which helps improve the quality of access to reliable news (Walunj et al. 2023). Additionally, proper news classification based on topics significantly aids in accurately detecting fake news and prevents the spread of fraudulent information and the promotion of false news. In particular, in the era of spreading rumors and fake news in cyberspace, news classification can help to distinguish between true and false news and to make effective decisions to deal with these phenomena (Gangwar and Ravi 2022).

One effective approach to processing and classifying large amounts of data is through the use of deep learning models in natural language processing. These models can learn from large

datasets and detect linguistic and semantic patterns in texts. However, deep learning models require managing numerous challenges, including model selection, tuning convergence parameters, lengthy training, testing more complex cases, and computational solid hardware platforms (Nadeem et al. 2023).

In contrast, classic ML models are simpler and more cost-effective than deep learning models. They usually use fewer parameters and data for training and are efficient for many NLP tasks. Another advantage of classic ML models is their high speed and fast processing power (Liao et al. 2022). They can be efficiently run on systems with limited resources and quickly provide reasonable responses, and this is particularly important when evaluating and processing real-time data or massive datasets (Ameer et al. 2023).

Therefore, the current study examines the performance of four classic learning models in news text classification, including NB, RF, SVM, and LR. For this purpose, the BBC dataset is used, which includes more than 2100 news texts categorized into five categories: business, entertainment, politics, sports, and technology. Same as manual numerical simulations, since classic models lack implicit preprocessing, several explicit preprocessing layers are used in the current study, including converting all letters to lowercase, removing punctuation, tokenization, and removing stop words to prepare the raw data (Farsad et al. 2022). After training the mentioned models, various metrics such as accuracy, confusion matrix, and mean square error are used to evaluate their performance accurately. This evaluation helps to draw a suitable conclusion about the accuracy and cost-effectiveness of these models compared to deep learning models.

The remaining of this study are divided into the following sections. First, the Related Work describes the prior studies, and how each study build up to for the current investigation. Next, Dataset section briefly describes the sources used to train the ML models. In the Preprocessing chapter, all the necessary operations implemented on the raw data for the training session are described. Subsequently, the Methodology section provides details regarding the utilized ML models. In the Results and Discussion, the model predictions are compared against each other and a careful performance analysis is performed. Lastly, the research highlights are summarized is Conclusion.

2. Related Work

Given the importance of accurate news classification, numerous studies have recently been conducted on this topic over the past decade. Among them, one can refer to the research on detecting fake news using a simple MBN (Granik and Mesyura 2017). This study presents a straightforward method for detecting fake news using a basic Bayes classifier. The method was implemented as a software system and tested against Facebook news posts. The model obtained an accuracy of around 74% in the test set, which is a commendable result considering the model's simplicity.

In the meantime, multi-label classification through RF has also attracted many researchers and has yielded outstanding results in news classification. For instance, a multi-label classification method based on RF, which can effectively discover correlated labels for optimizing label subset partitioning, was used in a study by (Liu et al. 2015). This method was tested on ten datasets and achieved a performance range of 70 to 85%, which is higher than other advanced multi-label classification algorithms.

Another study focused on detecting fake news from the IRJET dataset again uses LR to automatically identify fake content in news articles, which is essentially a binary classification. Like the current study, this research also used preprocessing functions like tokenization,

stemming, and exploratory data analysis such as response variable distribution and data quality check (i.e., zero or missing values) and achieved an accuracy of over 85% in most tests (Nada et al. 2008). It is also worth mentioning that LR, with the help of methods such as feature selection based on maximum penalized probability, has also been used in research such as this one, which reached higher accuracy in the discussion of news classification with less data (Abramovich, Grinshtein, and Levy 2021).

Additionally, numerous studies have examined the capability of SVM-based classifiers for news classification. For example, a quantitative analysis between SVM-based classifiers and studies on twin SVMs on news data is presented in a study by (Saigal and Khanna 2020), with the first version being evaluated as the more successful model.

Finally, comparisons between ML models have been conducted in various studies. For instance, in another study by (Singh et al. 2021), a news classification system using NB and SVM algorithms was implemented. Similar to the current study, this research classifies news into business, entertainment, politics, sports, and technology categories. This system helps users find their desired news articles with an accuracy of 87%, saving time and reducing news overload, demonstrating the success of classic ML models in text data classification.

By summarizing the points explained in previous studies, the present study aims to evaluate four of the most successful ML models in text data classification. It examines their capabilities and maximum achievable accuracy using various assessment metrics in the selected dataset. It is important to note that, like previous studies, different preprocessing layers have also been used to improve the implemented models, which will be detailed in subsequent sections.

3. Dataset

Nam The BBC news dataset is a unique collection of news articles compiled from a large archive of 2226 rows of news on this media (Greene and Cunningham 2006). Generally, this media is known as one of the largest and most reputable Western news networks, and this dataset represents the range and diversity of news produced by this media. The dataset includes articles categorized into the following five topics:

- Business: Articles related to economic subjects, markets, companies, and trade.
- Entertainment: News and articles about culture, art, music, cinema, and television.
- Politics: Articles covering political events, and international relations.
- Sports: News and reports on sports events, games, and competitions.
- Technology: Articles related to innovations, advancements, and challenges in information technology and other technological fields.

In summary, the data structure in this dataset includes two parts: the news text and its category, separated by a comma. Figure 1 shows the distribution of news in each category. Generally, the data, including texts, have been collected without any editing or filtering; therefore, they need preprocessing and cleaning to be ready for ML models. It is also worth mentioning that, in the current study, 20% of the data has been set aside for testing, and the remaining texts have been used for training the model.

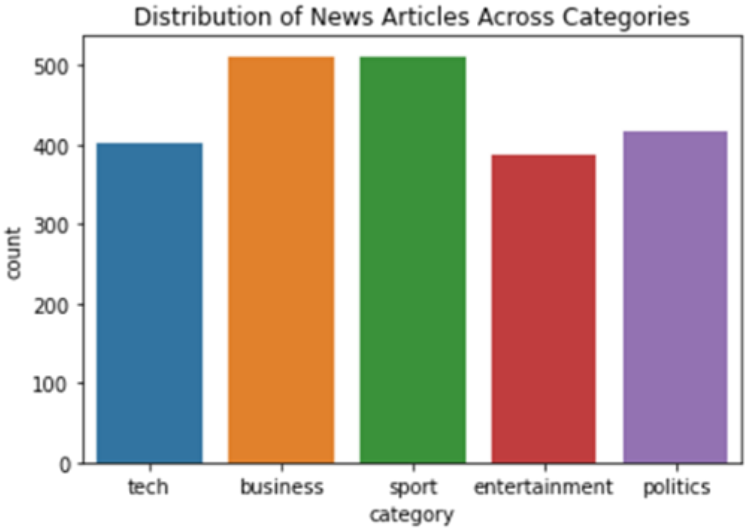

Figure 1: The Category distribution in the BBC news dataset

4. Preprocessing

Text preprocessing is a crucial step in NLP for several key reasons (Luo and Chong 2020). Firstly, appropriate preprocessing ensures that raw text data are standardized and structured, eliminating inconsistencies and noise (Petukhova and Fachada 2022). This consistency is vital for improving the performance of ML models, enabling them to detect patterns and relationships more effectively. Secondly, preprocessing simplifies and enhances the interpretability of the text by removing punctuation, converting to lowercase, tokenizing, and eliminating irrelevant stop words, making it simpler and more informative (Urane and Deshpande 2022). This simplified text is beneficial for a wide range of NLP tasks. By converting text into numerical representations, it bridges the gap between natural language and machine-understandable data, allowing algorithms to process the text, recognize patterns, and make data-based decisions (Nesca et al. 2022). In the current study, the preprocessing methods described following have been employed.

4.1. Text Standardization

In the field of text preprocessing, one of the fundamental steps is converting all words in a text to a standard format (Elov, Khamroeva, and Xusainova 2023). This seemingly simple transformation significantly affects the learning model's performance. In fact, by converting all text to a specified and standard format of characters, ML models can recognize that words with upper-case, lower-case, and mixed-case letters are essentially the same. This process smoothens text handling and reduces the number of unknown entities in the text (Bouaine, Benabbou, and Sadgali 2023). For example, words like "bOOk" and "BOOKS" should be standardized to a standard form, "Book", so the model can accurately recognize that they have similar meanings. This process paves the way for consistent text analysis and reduces the model's sensitivity to minor variations (Badawi et al. 2023). It also increases operational speed and reduces the memory the models require (Balouch and Hussain 2023). The current study considers the singular form and all lowercase letters as the standard format.

4.2. Punctuation Removal

The next step in text preprocessing involves the removal of punctuation marks, including periods, commas, dashes, question marks, exclamation marks, and numbers from the text. This removal is essential to ensure that models focus solely on the textual content. However, it should be noted that in certain specific preprocessing scenarios, each punctuation mark is

replaced by a specific word to assist the model in better understanding the text when necessary (Kemala and Shiddiqi 2023). Although this approach can create other problems in the presence of abbreviations and various punctuation marks in the text, it is generally not used in most NLP applications. Consequently, the current study has identified and removed all punctuation marks.

4.3. Tokenization

Tokenization is an essential step in text preprocessing, where the text is divided into smaller units, usually referred to as tokens (Hiraoka et al. 2020). These tokens can be individual words or subunits. Each word is converted into a token, breaking the text into its most basic elements (Yerpude, Jakhotiya, and Chandak 2015). For example, the phrase "we are students" is converted into three separate tokens: "we", "are" and "students". Tokenization is a basis for various NLP tasks and facilitates ML models in understanding and manipulating textual data.

4.4. Stop Words Removal

The removal of stop words usually occurs at this stage of text preprocessing. Stop words are words that usually have minimal value in the context of text classification or analysis (Ramdhani et al. 2020). Examples of stop words include: "and", "at", "to". Removing these words simplifies the text and makes it more meaningful for subsequent analysis. Stop words are prevalent in various texts and can overshadow more meaningful and impactful words if not addressed (Kale et al. 2023). By eliminating these common stop words, the focus of text analysis shifts towards more informative and relevant terms, ultimately enhancing the quality of text processing and classification (Ladani and Desai 2020). The lists all the stop words removed in preprocessing step is also provided in the Appendix.

4.5. Numerical Representation of Text

After completing the initial preprocessing stages, the transformed texts are represented in a numerical format that ML models can understand (Dang, Moreno-García, and De la Prieta 2020). This step is crucial for enabling models to operate on textual data effectively. One commonly used method for this goal is the Term Frequency-Inverse Document Frequency (TF-IDF) model (Ahuja et al. 2019). This method's mission is to provide a numerical representation used in NLP and information retrieval to evaluate the importance of a term in a document or a collection of documents (Das, Kamalanathan, and Alphonse 2021).

The TF component measures the number of times a term appears in a document (Pietro 2020). It represents the local importance of a term in a specific document, which is calculated as follows:

$$TF(t,d) = \frac{Number\ of\ times\ term\ t\ appears\ in\ document\ d}{Total\ Number\ of\ Terms\ in\ Document\ d} \quad (1)$$

Formula 1 uses *t'* to represent the term, and *'d'* to represent the document. TF normalizes the frequency of a term in a document by dividing it by the total number of terms in the document. This normalization prevents bias towards longer documents (Soufyane, Abdelhakim, and Ahmed 2021). The IDF component determines how often a term appears in the entire set of documents. Terms that appear in many documents are considered less distinctive and, therefore, less valuable for classification or retrieval (Christian, Agus, and Suhartono 2016). Formula 2 is used to calculate this component:

$$IDF(t) = log\left(\frac{Total\ number\ of\ documents\ in\ the\ corpus}{Number\ of\ documents\ containing\ term\ t}\right) \quad (2)$$

IDF calculates the logarithm of total docs to docs with term. The logarithm diminishes the impact of prevalent terms while emphasizing rarer terms (Assayed, Shaalan, and Alkhatib 2023). Finally, to calculate the TF-IDF score for a term in a specific document, Formula 3 is used:

$$TF - IDF(t, d) = TF(t, d)\, IDF(t) \tag{3}$$

The TF-IDF score provides a balanced assessment of the importance of a term in a document and across a collection of documents. Terms with higher scores are common in a specific document but relatively rare across the entire collection, making them more distinctive and meaningful for tasks such as document classification, information retrieval, and text mining (Rameshbhai and Paulose 2019).

For example, if the word "car" frequently appears in one document but rarely in others, the TF-IDF score assigns a higher weight to "car" in the document where it is more prominent. This numerical representation equips ML models to recognize patterns and connections within texts and perform accurate classification (Fattahi and Mejri 2021). The TF-IDF metric and similar techniques bridge the gap between natural language and machine-understandable data, facilitating advanced text analysis and classification (Noersasongko et al. 2021).

5. Methodology

Selecting a model, evaluation metrics, and methods for training and testing in ML are of particular importance and can significantly impact the efficiency and performance of a ML system (Chai et al. 2023). Each ML model has its own characteristics and capabilities, and a suitable model may exist for each problem. Incorrect model selection can lead to unstable and unpredictable results (Pan et al. 2022). Therefore, the model selection process, considering the problem's nature and the data type, is crucial (Mohammad Hossein Zolfagharnasab et al. 2020).

Moreover, choosing a suitable model can lead to more efficient use of time and computational resources. Different models may require varying amounts of data and time for training. Choosing a model inconsistent with the problem's nature can lead to lengthy training and a waste of computational resources. Hence, selecting an appropriate model as a fundamental step in the ML process cannot be overlooked and requires careful consideration and experience in making this choice (Jeevaraj et al. 2023). The following will examine four models used in this research, which are MNB, LR, RF, and SVM.

5.1. Multinomial Naive Bayes

The NB model is a popular classification algorithm used in NLP and many other ML tasks (Ige and Adewale 2022). This algorithm is suitable for text classification tasks like spam detection, sentiment analysis, and topic categorization, as it is based on Bayes' theorem. (Bahri, Saputra, and Wajhillah 2017).

The key idea behind the NB model is the independence of features from each other. The term 'Naive' in its name refers to the simplifying assumption of feature independence despite the potential interdependencies, assessing the effect of each feature relative to the target. This simplistic hypothesis might seem overly naive at first glance, but it has been computationally efficient and surprisingly successful in text classification tasks. The NB model calculates the probability of a document belonging to a particular class based on the occurrence of words in the document, using Formula 4:

$$P(Class_j|X_i) = P(Class_j)\prod_{i=1}^{n} P(X_i|Class_j) \quad (4)$$

For classification decision-making, the Bayes model first calculates *P(Class|X)* for each possible class and selects the class with the highest probability. During training, the NB model must learn the conditional probability of each word given the class *P*(*X*|Class) from a labeled dataset. Then, the model uses these probabilities to predict new, unlabeled documents. Despite its simplifying assumption of independence, it often works well for language processing tasks, especially where limited training data is available.

5.2. Logistic Regression

The LR model is a ML model used for classification problems, specifically in NLP for detecting and classifying texts (Zhang 2023). This model utilizes logistic (or sigmoid) distribution to calculate the probability of the dependent variable given the features (which can be words or tokens) for classifying inputs (Lineback et al. 2021). The logistic function formula is calculated as per Formula 5:

$$P(Class_j|X_i) = P(\frac{1}{1+\exp{(\beta X)}}) \quad (5)$$

For classification decision-making, the LR model first calculates *P*(Class|*X*) for each possible class and classifies the document as positive if the estimated probability is higher than a certain threshold (usually 0.5). Otherwise, it is classified as negative. In summary, LR is a fundamental classification algorithm that models the probability of a document belonging to a particular class and ensures that the output is a probability, making it interpretable and suitable for classification decision-making.

5.3. Random Forest

RF is a robust ML algorithm that is used for classification and regression tasks. It is beneficial for handling high-dimensional data and is suitable for NLP applications. Unlike linear models, RF does not have equations, but its basic principles can be divided into three parts. RF is known for its stability and accuracy, making it a popular choice for many data scientists and ML practitioners. First, RF uses decision trees as base models. Decision trees are built using random subsets of training data and features/words (Turner et al. 2017). The final decision is made by combining the predictions of all individual trees, which helps increase accuracy and generalizability. Second, RF employs a bagging technique, which involves creating several subsets of the training data with replacement. Each decision tree is trained on a different subset of data, introducing a unique variety to the model. Third, RF uses feature sub-setting to introduce randomness and reduce overfitting. In each split in the decision tree, only a random subset of features is considered. This significantly reduces the model's dependence on any particular feature (Antony Vijay, Anwar Basha, and Arun Nehru 2021). Fourth, they make predictions for new data after training all decision trees. For classification tasks, the class labels predicted by each tree are aggregated (voting), and the class with the most votes is selected as the final prediction.

5.4. Support Vector Machine

SVM is a supervised ML algorithm used in classification and regression tasks. It is particularly useful in solving classification problems where the goal is to identify the class or category to which a data point belongs. The basic concept of SVM is to find a hyperplane that can optimally separate the data points of different classes in a high-dimensional feature space while

maximizing the margin between the two classes. (Suvarna Lakshmi, Saxena, and Kumar 2023). Abstractly, this hyperplane in two-dimensional space is the separating line, but in higher dimensions and with more features, it becomes a multi-dimensional surface. Formula 6 presents such multi-dimensional hyper-planes, where W represents the weight vector, X is the feature vector, and b is the bias.

$$W^T X + b = 0 \tag{6}$$

In SVM, the goal is to maximize the distance between the hyperplane and the closest data point (support vector) of each class, given a regularization term that prevents overfitting, as written in Formula 7:

$$\begin{aligned} &Minimize: \frac{1}{2} W^T W + C \sum_{i=1}^{n} \zeta \\ &Subject\ to: y(W^T \emptyset(x) + b) \geq 1 - \zeta \end{aligned} \tag{7}$$

The decision boundary is then determined by Formula 6. An interesting aspect of SVM is the kernel trick, which allows the model to handle non-linear data. A kernel function like ϕ allows data to be transformed into a higher-dimensional space, which might be linearly separable (Sayan Majumder, Anuran Aich, and Satrajit Das 2022).

In the current study, the Radial Basis Function (RBF) model has been used, which is essentially a Gaussian function as per Formula 8:

$$\emptyset(x) = exp\left(-\gamma \frac{\|x - x'\|^2}{2}\right) \tag{8}$$

Additionally, in some cases, data overlap makes it impossible to find a hyperplane that entirely separates the classes. Therefore, a slack variable is used to ignore some of these variables for greater model comprehensiveness (Tohira et al. 2022).

5.5. Evaluation Methods

The importance of evaluation metrics especially for DL simulations is due to the complex and non-linear nature of these models, which makes them less explainable (black-box) from human perspective (M. H. Zolfagharnasab et al. 2021). Therefore, proper evaluation metrics for ML models provide precise and quantitative information about the models' capabilities in the prediction, classification, or regression of data. For a wide range of ML studies, metrics such as accuracy, sensitivity, and specificity are used to guide the data engineers toward selecting valuable features, detecting model weak spots, and perform parameter tunning. Such information guides ML engineers toward selecting valuable features, detecting model weak spots, perform parameter tunning, and obtain a more in-depth conclusions for the model selection, which is the key element in any ML related study.

Similarly, the current study also selects several evaluation metrics based on the confusion matrix, such as accuracy, precision, and recall. The confusion matrix effectively displays the difference between predictions and reality, allowing users to discern which categories have been correctly classified and which have been confused with each other. This matrix can reveal the strengths and weaknesses of the model. Each metric represents different aspects of model

performance and can guide the selection of the best model for a specific task. Additionally, the current study has used the Root Mean Square Error (RMSE) metric to provide a better understanding of the size of errors.

It should be noted that since RMSE is suitable for continuous data and regression problems, numerical mapping has been used in this project to convert classification data into a continuous space to apply this metric.

6. Results and Discussion

This study evaluated the performance of four ML models - MNB, LR, RF, and SVM in a multiclass news classification task. The classification results of each model, presented through accuracy, recall, and F1 score metrics, are analyzed in detail. Additionally, details related to the confusion matrix are displayed in Figure 2.

According to Table 1, the NB model shows a high level of accuracy, reaching 96.6%. The accuracy, recall and F1 score metrics are outstanding across all news categories, with notable performance in classifying entertainment and politics categories. However, the business category has a lower recall metric compared to other models. In summary, this model provides a reliable and consistent performance for classifying news data.

Table 2 reveals that the LR model, with a stunning accuracy of 97.1%, exhibits much higher precision, recall, and F1 scores in all categories, particularly in the "business" category. Compared to NB and SVM, its recall is evaluated lower, especially in the "politics" category. However, LR in this application shows competitive performance with much stronger models.

Table 3 shows that while the RF model has the weakest accuracy at 94%, it still provides acceptable results. An interesting point about this model is the balanced performance of all accuracy, recall, and F1 score metrics across all news categories, indicating the reliability of this model. Due to its decision tree ensemble approach, it offers a good balance between accuracy and interpretability.

According to Table 4, the SVM stands out as the best-performing model with the highest accuracy of 98%. In this model, almost all accuracy, recall, and F1 score metrics are perfect in all categories, although the business and political news consistently seem to have the highest successful separation rates from other categories.

In summary, the SVM emerges as the most successful model for multiclass news classification, providing the highest accuracy and F1 scores. However, both NB and LR also show strong performance and are suitable alternatives. The RF, while slightly less accurate, provides a balanced approach. Nonetheless, it is beneficial to examine the reasons behind the performance of the models based on their characteristics.

LR, a classification algorithm, works well for data sets with many features and where a linear relationship exists between inputs and outputs. Given that we used the TF-IDF model to convert texts into numerical vectors, our texts are placed in a space with many features. In such cases, LR can perform very well.

The MNB model, based on Bayes' law, assumes that features are independent of each other. While this assumption is not true in many real-world cases, NB performs adequately in text classification tasks. However, the assumption of feature independence can cause inaccuracies in some instances.

RF, a decision tree-based algorithm, makes predictions by building several decision trees and combining their results. It is more robust against outliers and noise in data and usually performs well in many tasks. However, it might not perform as well as LR with text data transformed into long vectors with many features.

Finally, the outstanding performance of the SVM can be attributed to addressing non-linear data patterns. Using various kernel functions, like polynomial and radial basis functions, this model can map data into higher-dimensional spaces, making it particularly effective for classifying data with non-linear decision boundaries. Also, the SVM can perform classification tasks effectively in dealing with high-dimensional feature spaces typical of natural language applications. This is significant because, due to its robustness against noise, support vectors prioritize points near the decision boundary, making outliers less impactful on the model.

Given the above explanations, it can be understood why, in this specific case, the SVM outperformed the other three models. The nature of the data and the way they are transformed into numerical features play a crucial role in determining each model's performance. In such cases, choosing the suitable model for the specific task is key to success.

	Precision	Recall	F1-score
Technology	0.96	0.94	0.95
Economy	1.00	0.91	0.95
Sports	0.93	0.99	0.96
Entertainment	0.99	1.00	0.99
Politics	0.95	0.99	0.97
Model Accuracy	0.996		
RMSE	0.402		

Table 1: Evaluation Metrics of the MNB model

	Precision	Recall	F1-score
Technology	0.95	0.94	0.95
Economy	1.00	0.95	0.97
Sports	0.94	0.99	0.96
Entertainment	0.98	1.00	0.99
Politics	0.99	0.98	0.98
Model Accuracy	0.971		
RMSE	0.373		

Table 2: Evaluation Metrics of the LR model

	Precision	Recall	F1-score
Technology	0.90	0.93	0.92
Economy	1.00	0.90	0.95
Sports	0.93	0.95	0.94
Entertainment	0.96	0.99	0.97
Politics	0.93	0.93	0.93
Model Accuracy	0.944		
RMSE	0.586		

Table 3: Evaluation Metrics of the RF model

	Precision	Recall	F1-score
Technology	0.98	0.94	0.96
Economy	1.00	0.99	0.99
Sports	0.94	0.99	0.96
Entertainment	0.98	1.00	0.99
Politics	1.00	0.99	0.99
Model Accuracy	0.98		
RMSE	0.280		

Table 4: Evaluation Metrics of the SVM model

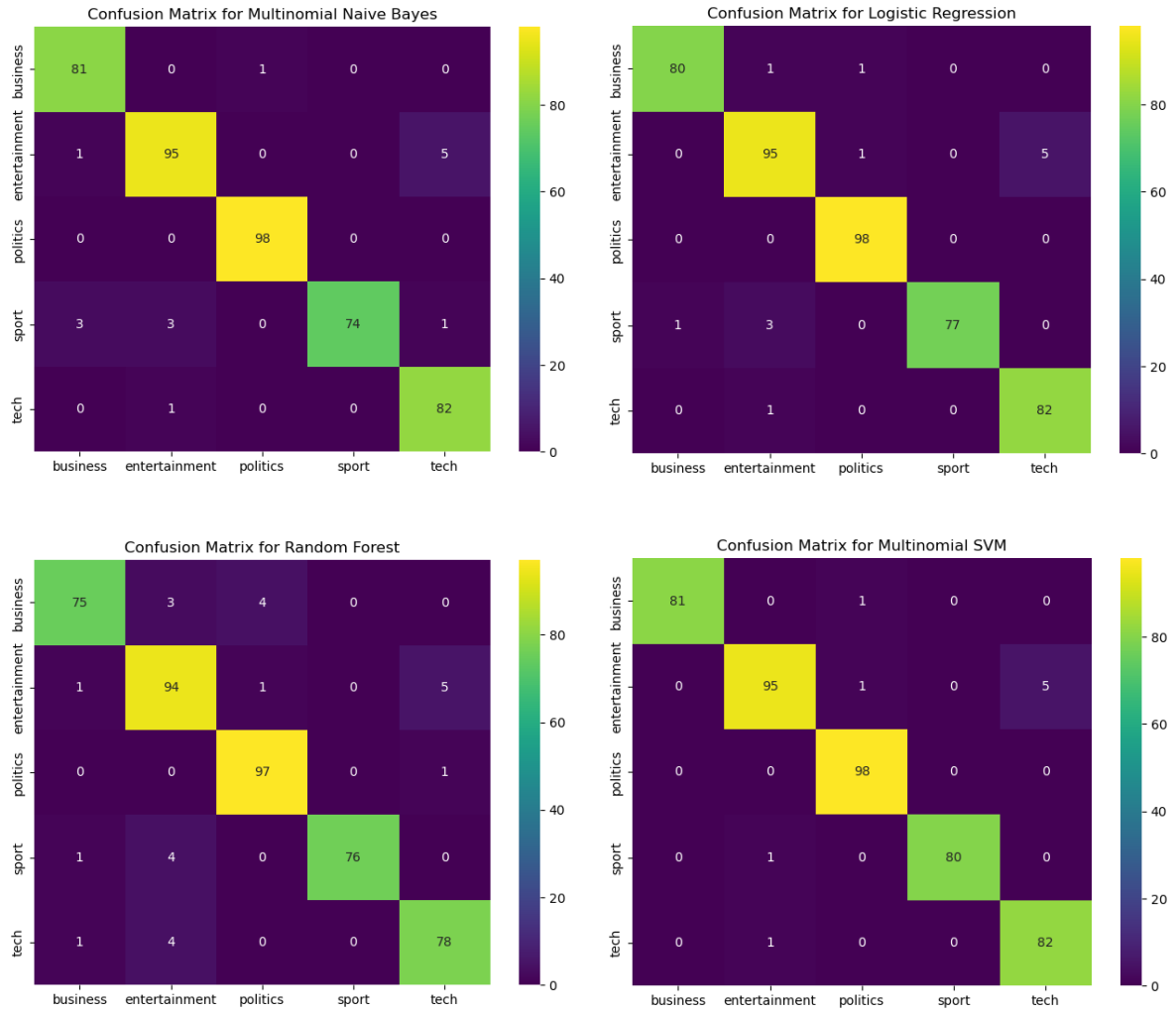

Figure 2: Confusion matrix of the performance of the models implemented to classify the BBC news dataset

7. Conclusion

The relentless flow of news underscores the need for thorough evaluation to guarantee accurate and timely delivery of information to its intended audience. This necessity highlights the role of NLP models to efficiently process and interpreting vast amounts of textual data, ensuring both the accuracy and prompt dissemination of news.

As a response to the noted issue, this study used the BBC news dataset to evaluate resource-efficient classic ML techniques for news categorization. By examining the four models of NB, RF, SVM, and LR, it is found that the SVM has the best performance with an accuracy of 98%. The results suggest that with the right approach in preprocessing, modeling, and evaluation, text classification can achieve remarkably high accuracy and efficiency. Therefore, the utilization of advanced models and extensive hardware is not the only path to harnessing the power of artificial intelligence in text classification.

References

Abramovich, Felix, Vadim Grinshtein, and Tomer Levy. 2021. “Multiclass Classification by Sparse Multinomial Logistic Regression.” *IEEE Transactions on Information Theory* 67 (7). https://doi.org/10.1109/TIT.2021.3075137.

Ahuja, Ravinder, Aakarsha Chug, Shruti Kohli, Shaurya Gupta, and Pratyush Ahuja. 2019. "The Impact of Features Extraction on the Sentiment Analysis." In *Procedia Computer Science*. Vol. 152. https://doi.org/10.1016/j.procs.2019.05.008.

Ameer, Iqra, Necva Bölücü, Muhammad Hammad Fahim Siddiqui, Burcu Can, Grigori Sidorov, and Alexander Gelbukh. 2023. "Multi-Label Emotion Classification in Texts Using Transfer Learning." *Expert Systems with Applications* 213. https://doi.org/10.1016/j.eswa.2022.118534.

Antony Vijay, J., H. Anwar Basha, and J. Arun Nehru. 2021. "A Dynamic Approach for Detecting the Fake News Using Random Forest Classifier and Nlp." In *Advances in Intelligent Systems and Computing*. Vol. 1257. https://doi.org/10.1007/978-981-15-7907-3_25.

Assayed, Suha K., Khaled Shaalan, and Manar Alkhatib. 2023. "A Chatbot Intent Classifier for Supporting High School Students." *EAI Endorsed Transactions on Scalable Information Systems* 10 (3). https://doi.org/10.4108/eetsis.v10i2.2948.

Badawi, Soran, Ari M. Saeed, Sara A. Ahmed, Peshraw Ahmed Abdalla, and Diyari A. Hassan. 2023. "Kurdish News Dataset Headlines (KNDH) through Multiclass Classification." *Data in Brief* 48. https://doi.org/10.1016/j.dib.2023.109120.

Bahri, Saeful, Rizal Amegia Saputra, and Rusda Wajhillah. 2017. "Analisa Sentimen Berbasis Natural Languange Processing (NLP) Dengan Naive Bayes Clasifier." *Konferensi Nasional Ilmu Sosial & Teknologi* 1 (1).

Balouch, Bilal Ahmed Khan, and Fawad Hussain. 2023. "A Transformer Based Approach for Abstractive Text Summarization of Radiology Reports." *International Conference on Applied Engineering and Natural Sciences* 1 (1). https://doi.org/10.59287/icaens.1042.

Bouaine, Chaimaa, Faouzia Benabbou, and Imane Sadgali. 2023. "Word Embedding for High Performance Cross-Language Plagiarism Detection Techniques." *International Journal of Interactive Mobile Technologies* 17 (10). https://doi.org/10.3991/ijim.v17i10.38891.

Chai, Chengliang, Jiayi Wang, Yuyu Luo, Zeping Niu, and Guoliang Li. 2023. "Data Management for Machine Learning: A Survey." *IEEE Transactions on Knowledge and Data Engineering* 35 (5). https://doi.org/10.1109/TKDE.2022.3148237.

Christian, Hans, Mikhael Pramodana Agus, and Derwin Suhartono. 2016. "Single Document Automatic Text Summarization Using Term Frequency-Inverse Document Frequency (TF-IDF)." *ComTech: Computer, Mathematics and Engineering Applications* 7 (4). https://doi.org/10.21512/comtech.v7i4.3746.

Dang, Nhan Cach, María N. Moreno-García, and Fernando De la Prieta. 2020. "Sentiment Analysis Based on Deep Learning: A Comparative Study." *Electronics (Switzerland)* 9 (3). https://doi.org/10.3390/electronics9030483.

Das, Mamata, Selvakumar Kamalanathan, and Pja Alphonse. 2021. "A Comparative Study on TF-IDF Feature Weighting Method and Its Analysis Using Unstructured Dataset." In *CEUR Workshop Proceedings*. Vol. 2870.

Elov, B. B., Sh. M. Khamroeva, and Z. Y. Xusainova. 2023. "The Pipeline Processing of NLP." *E3S Web of Conferences* 413. https://doi.org/10.1051/e3sconf/202341303011.

Farsad, Saeed, Mahmoud Mashayekhi, Mohammad Hossein Zolfagharnasab, Mohammad Lakhi, Foad Farhani, Kourosh Zareinia, and Vahab Okati. 2022. "The Effects of Tube Dimples-Protrusions on the Thermo-Fluidic Properties of Turbulent Forced-Convection." *Case Studies in Thermal Engineering* 35. https://doi.org/10.1016/j.csite.2022.102033.

Fattahi, Jaouhar, and Mohamed Mejri. 2021. “SpaML: A Bimodal Ensemble Learning Spam Detector Based on NLP Techniques.” In *2021 IEEE 5th International Conference on Cryptography, Security and Privacy, CSP 2021*. https://doi.org/10.1109/CSP51677.2021.9357595.

Gangwar, Akhilesh Kumar, and Vadlamani Ravi. 2022. “A Novel BGCapsule Network for Text Classification.” *SN Computer Science* 3 (1). https://doi.org/10.1007/s42979-021-00963-4.

Granik, Mykhailo, and Volodymyr Mesyura. 2017. “Fake News Detection Using Naive Bayes Classifier.” In *2017 IEEE 1st Ukraine Conference on Electrical and Computer Engineering, UKRCON 2017 - Proceedings*. https://doi.org/10.1109/UKRCON.2017.8100379.

Greene, Derek, and Pádraig Cunningham. 2006. “Practical Solutions to the Problem of Diagonal Dominance in Kernel Document Clustering.” In *ACM International Conference Proceeding Series*. Vol. 148. https://doi.org/10.1145/1143844.1143892.

Hiraoka, Tatsuya, Sho Takase, Kei Uchiumi, Atsushi Keyaki, and Naoaki Okazaki. 2020. “Optimizing Word Segmentation for Downstream Task.” In *Findings of the Association for Computational Linguistics Findings of ACL: EMNLP 2020*. https://doi.org/10.18653/v1/2020.findings-emnlp.120.

Husin, Nanang. 2023. “Komparasi Algoritma Random Forest, Naïve Bayes, Dan Bert Untuk Multi-Class Classification Pada Artikel Cable News Network (CNN).” *Jurnal Esensi Infokom : Jurnal Esensi Sistem Informasi Dan Sistem Komputer* 7 (1). https://doi.org/10.55886/infokom.v7i1.608.

Ige, Tosin, and Sikiru Adewale. 2022. “AI Powered Anti-Cyber Bullying System Using Machine Learning Algorithm of Multinomial Naïve Bayes and Optimized Linear Support Vector Machine Interception of Cyberbully Contents in a Messaging System by Machine Learning Algorithm.” *International Journal of Advanced Computer Science and Applications* 13 (5). https://doi.org/10.14569/IJACSA.2022.0130502.

Kale, Sunil D., Parikshit N. Mahalle, Renu Kachhoria, Santosh Kumar, Prasad Chaudhari, and Vivek D. Patil. 2023. “Marathi Text Summarization through NLP and Deep Learning Mechanism.” *Journal of Autonomous Intelligence* 6 (3). https://doi.org/10.32629/jai.v6i3.1009.

Kemala, Ade Putera, and Hafizh Ash Shiddiqi. 2023. “Analysis of Indonesian Language Dataset for Tax Court Cases: Multiclass Classification of Court Verdicts.” *Jurnal Riset Informatika* 5 (3). https://doi.org/10.34288/jri.v5i3.555.

Ladani, Dhara J., and Nikita P. Desai. 2020. “Stopword Identification and Removal Techniques on TC and IR Applications: A Survey.” In *2020 6th International Conference on Advanced Computing and Communication Systems, ICACCS 2020*. https://doi.org/10.1109/ICACCS48705.2020.9074166.

Liao, Qing, Heyan Chai, Hao Han, Xiang Zhang, Xuan Wang, Wen Xia, and Ye Ding. 2022. “An Integrated Multi-Task Model for Fake News Detection.” *IEEE Transactions on Knowledge and Data Engineering* 34 (11). https://doi.org/10.1109/TKDE.2021.3054993.

Lineback, Christina M., Ravi Garg, Elissa Oh, Andrew M. Naidech, Jane L. Holl, and Shyam Prabhakaran. 2021. “Prediction of 30-Day Readmission After Stroke Using Machine Learning and Natural Language Processing.” *Frontiers in Neurology* 12. https://doi.org/10.3389/fneur.2021.649521.

Liu, Feng, Xiaofeng Zhang, Yunming Ye, Yahong Zhao, and Yan Li. 2015. “MLRF: Multi-Label Classification through Random Forest with Label-Set Partition.” In *Lecture Notes in*

Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Vol. 9227. https://doi.org/10.1007/978-3-319-22053-6_44.

Luo, Jack W., and Jaron J.R. Chong. 2020. "Review of Natural Language Processing in Radiology." *Neuroimaging Clinics of North America*. https://doi.org/10.1016/j.nic.2020.08.001.

Nada, Fathima, Bariya Firdous Khan, Aroofa Maryam, and Zameer Ahmed. 2008. "Fake News Detection Using Logistic Regression." *International Research Journal of Engineering and Technology* 5577 (May).

Nadeem, Muhammad Imran, Kanwal Ahmed, Dun Li, Zhiyun Zheng, Hafsa Naheed, Abdullah Y. Muaad, Abdulrahman Alqarafi, and Hala Abdel Hameed. 2023. "SHO-CNN: A Metaheuristic Optimization of a Convolutional Neural Network for Multi-Label News Classification." *Electronics (Switzerland)* 12 (1). https://doi.org/10.3390/electronics12010113.

Nesca, Marcello, Alan Katz, Carson K. Leung, and Lisa M. Lix. 2022. "A Scoping Review of Preprocessing Methods for Unstructured Text Data to Assess Data Quality." *International Journal of Population Data Science*. https://doi.org/10.23889/ijpds.v7i1.1757.

Noersasongko, Edi, Guruh Fajar Shidik, Adhitya Nugraha, Pulung Nurtantio Andono, and Edi Jaya Kusuma. 2021. "Automatic Integration of Ubiquitous Access Address in Camera Surveillance System Using Natural Language Processing." *International Review on Modelling and Simulations* 14 (1). https://doi.org/10.15866/iremos.v14i1.19661

Pan, Jiaming, Xiao Jiang, Zean Tian, Yikun Hu, and Kenli Li. 2022. "ML Model Optimization-Selection and GFA Prediction for Binary Alloys." *Crystal Growth and Design* 22 (4). https://doi.org/10.1021/acs.cgd.1c01519.

Petukhova, Alina, and Nuno Fachada. 2022. "TextCL: A Python Package for NLP Preprocessing Tasks." *SoftwareX* 19. https://doi.org/10.1016/j.softx.2022.101122.

Pietro, Mauro Di. 2020. "Text Classification with NLP: Tf-Idf vs Word2Vec vs BERT." *Medium*.

Ramdhani, Muhammad Ali, Muhammad Ali Ramdhani, Dian Sa adillah Maylawati, and Teddy Mantoro. 2020. "Indonesian News Classification Using Convolutional Neural Network." *Indonesian Journal of Electrical Engineering and Computer Science* 19 (2). https://doi.org/10.11591/ijeecs.v19.i2.pp1000-1009.

Rameshbhai, Chaudhary Jashubhai, and Joy Paulose. 2019. "Opinion Mining on Newspaper Headlines Using SVM and NLP." *International Journal of Electrical and Computer Engineering* 9 (3). https://doi.org/10.11591/ijece.v9i3.pp2152-2163.

Saigal, Pooja, and Vaibhav Khanna. 2020. "Multi-Category News Classification Using Support Vector Machine Based Classifiers." *SN Applied Sciences* 2 (3). https://doi.org/10.1007/s42452-020-2266-6.

Sayan Majumder, Anuran Aich, and Satrajit Das. 2022. "SENTIMENT ANALYSIS OF PEOPLE DURING COVID- 19 USING SVM AND LOGISTIC REGRESSION ANALYSIS." *EPRA International Journal of Multidisciplinary Research (IJMR)*. https://doi.org/10.36713/epra10424.

Singh, Yash Veer, Piyush Naithani, Parvez Ansari, and Pragya Agnihotri. 2021. "News Classification System Using Machine Learning Approach." In *Proceedings - 2021 3rd International Conference on Advances in Computing, Communication Control and Networking, ICAC3N 2021*. https://doi.org/10.1109/ICAC3N53548.2021.9725409.

Soufyane, Ayanouz, Boudhir Anouar Abdelhakim, and Mohamed Ben Ahmed. 2021. "An Intelligent Chatbot Using NLP and TF-IDF Algorithm for Text Understanding Applied to the Medical Field." In *Advances in Science, Technology and Innovation*. https://doi.org/10.1007/978-3-030-53440-0_1.

Suvarna Lakshmi, C., Sameer Saxena, and B. Suresh Kumar. 2023. "Design Text Mining Classifier for Covid-19 by Using the Machine Learning Techniques." *International Journal of Intelligent Systems and Applications in Engineering* 11 (2s).

Tohira, Hideo, Judith Finn, Stephen Ball, Deon Brink, and Peter Buzzacott. 2022. "Machine Learning and Natural Language Processing to Identify Falls in Electronic Patient Care Records from Ambulance Attendances." *Informatics for Health and Social Care* 47 (4). https://doi.org/10.1080/17538157.2021.2019038.

Truică, Ciprian Octavian, and Elena Simona Apostol. 2023. "It's All in the Embedding! Fake News Detection Using Document Embeddings." *Mathematics* 11 (3). https://doi.org/10.3390/math11030508.

Turner, Clayton A., Alexander D. Jacobs, Cassios K. Marques, James C. Oates, Diane L. Kamen, Paul E. Anderson, and Jihad S. Obeid. 2017. "Word2Vec Inversion and Traditional Text Classifiers for Phenotyping Lupus." *BMC Medical Informatics and Decision Making* 17 (1). https://doi.org/10.1186/s12911-017-0518-1.

Urane, Kimaya, and Arati Deshpande. 2022. "Deep Learning Based Fake News Detection." *International Journal on Recent and Innovation Trends in Computing and Communication* 10 (7). https://doi.org/10.17762/ijritcc.v10i7.5578.

Walunj, Parmesh, Krupa Shah, Rishi Tank, Atharva Mathure, Ritesh Shekhar, and Ms. Deepali Kadam. 2023. "Tag Recommendation System for Marathi News Articles by Using Multi-Label Classification." *International Journal for Research in Applied Science and Engineering Technology* 11 (4). https://doi.org/10.22214/ijraset.2023.50626.

Yerpude, Prajakta, Rashmi Jakhotiya, and Manoj Chandak. 2015. "Algorithm for Text to Graph Conversion and Summarizing Using NLP: A New Approach for Business Solutions." *International Journal on Natural Language Computing* 4 (4). https://doi.org/10.5121/ijnlc.2015.4403.

Zhang, Hanwen. 2023. "MBTI Personality Prediction Based on BERT Classification." *Highlights in Science, Engineering and Technology* 34. https://doi.org/10.54097/hset.v34i.5497.

Zolfagharnasab, M. H., M. Salimi, H. Zolfagharnasab, H. Alimoradi, M. Shams, and C. Aghanajafi. 2021. "A Novel Numerical Investigation of Erosion Wear over Various 90-Degree Elbow Duct Sections." *Powder Technology*. https://doi.org/10.1016/j.powtec.2020.11.059.

Zolfagharnasab, Mohammad Hossein, Cyrus Aghanajafi, Soheil Kavian, Niloufar Heydarian, and Mohammad Hossein Ahmadi. 2020. "Novel Analysis of Second Law and Irreversibility for a Solar Power Plant Using Heliostat Field and Molten Salt." *Energy Science and Engineering*. https://doi.org/10.1002/ese3.802.

Zolfagharnasab, MohammadHossein, Mona Zamani Pedram, Siamak Hoseinzadeh, and Kambiz Vafai. 2022. "Application of Porous-Embedded Shell and Tube Heat Exchangers for The Waste Heat Recovery Systems." *Applied Thermal Engineering* 118452. https://doi.org/10.1016/j.applthermaleng.2022.118452.

Appendix

The list of stop words removed at the preprocess stage are given in the following.

{"a", "about", "above", "after", "again", "against", "all", "am", "an", "and", "any", "are", "aren't", "as", "at", "be", "because", "been", "before", "being", "below", "between", "both", "but", "by", "can't", "cannot", "could", "couldn't", "did", "didn't", "do", "does", "doesn't", "doing", "don't", "down", "during", "each", "few", "for", "from", "further", "had", "hadn't", "has", "hasn't", "have", "haven't", "having", "he", "he'd", "he'll", "he's", "her", "here", "here's", "hers", "herself", "him", "himself", "his", "how", "how's", "i", "i'd", "i'll", "i'm", "i've", "if", "in", "into", "is", "isn't", "it", "it's", "its", "itself", "let's", "me", "more", "most", "mustn't", "my", "myself", "no", "nor", "not", "of", "off", "on", "once", "only", "or", "other", "ought", "our", "ours", "ourselves", "out", "over", "own", "same", "shan't", "she", "she'd", "she'll", "she's", "should", "shouldn't", "so", "some", "such", "than", "that", "that's", "the", "their", "theirs", "them", "themselves", "then", "there", "there's", "these", "they", "they'd", "they'll", "they're", "they've", "this", "those", "through", "to", "too", "under", "until", "up", "very", "was", "wasn't", "we", "we'd", "we'll", "we're", "we've", "were", "weren't", "what", "what's", "when", "when's", "where", "where's", "which", "while", "who", "who's", "whom", "why", "why's", "with", "won't", "would", "wouldn't", "you", "you'd", "you'll", "you're", "you've", "your", "yours", "yourself", "yourselves"}