Text Classification in Bangla Using Machine Learning Approach

Tasnia Sharmin
Master's in Data science
University of New Haven
tshar8@ung.newhaven.edu

Mounika Davuluri
Master's in Data Science
University of New Haven

Abstract

With the increasing popularity of online platforms and the massive amount of textual data being generated, efficient text classification techniques are essential for organizing and understanding this data. In this project, we focus on classifying Bangla text comments into five categories: not bully, sexual, religious, threat, and troll. We explore the use of various machine learning algorithms, including Logistic Regression, SVM, Decision Tree, Random Forest, and Multinomial Naïve Bayes, to classify the comments based on their content. Our results show that SVM achieves the highest accuracy of 80.17%, followed by Logistic Regression with an accuracy of 78.34%. Decision Tree and Random Forest achieve accuracies of 74.24% and 75.42% respectively, while Multinomial Naïve Bayes achieves an accuracy of 70.83%. These findings highlight the effectiveness of machine learning algorithms in classifying Bangla text comments, with SVM emerging as the most suitable algorithm for this task. Our project contributes to the field of text classification in Bangla by providing insights into the performance of different machine learning algorithms on this task. The classification of Bangla text comments is crucial for online content moderation and sentiment analysis in Bangla-speaking communities. Future research could explore additional features or techniques to further improve classification accuracy and expand the application of these techniques to other languages and text classification tasks.

Introduction

In natural language processing (NLP), text classification is a fundamental task that entails classifying text into predetermined classes or categories. Effective text classification approaches are becoming more and more necessary due to the increasing amount of digital content available in multiple languages, including Bangla. Applications like sentiment analysis, spam identification, and topic modeling are made possible by machine learning (ML) techniques, which provide a strong foundation for automatically identifying text.

We use machine learning approaches to classify texts in the Bangla language in this work. Bangla is the official language of Bangladesh, also known as Bengali, the second most spoken language in India. Bangla is used widely, but there aren't as many resources available for text processing it as there are for languages like English, especially when it comes to machine learning. Due to the exponential increase in complicated texts and documents, a deeper comprehension of machine learning techniques is now required for precise text classification in a variety of applications. Several machine learning techniques have produced impressive outcomes in natural language processing; these techniques rely on their capacity to understand intricate models and non-linear data correlations. Finding appropriate architectures, structures, and methods for text classification is still a problem for scholars [1].

Investigating and creating machine learning models that can precisely classify Bangla text into several categories or classes is the aim of this research. To create and assess our text classification system, we make use of pre-existing Bangla text datasets and a variety of machine learning (ML) techniques, including support vector machines (SVM), Naive Bayes. In doing so, we hope to further the development of natural language processing (NLP) methods for the Bangla language, which will facilitate the efficient processing and comprehension of Bangla text data.

Related Work

Several studies have delved into text classification across various languages, including Bangla, employing diverse machine learning techniques. In today's digital era, technology plays a pivotal role, generating a vast amount of online textual content, notably news articles, where categorization is crucial for efficient access. While other languages have seen advancements, Bangla's categorization remains relatively undeveloped. To address this, one study applied machine learning classifiers and neural networks to categorize Bangla news articles, with LSTM performing the best. This study highlights the importance of automated categorization in managing the overwhelming volume of online content, especially in languages with limited resources. The results demonstrate the effectiveness of machine learning in improving the accessibility of Bangla news articles for online readers [2]. Another paper investigates the effectiveness of four supervised machine learning methods—Decision Tree (C 4.5), K-Nearest Neighbor (KNN), Naïve Bayes (NB), and Support Vector Machine (SVM)—for categorizing Bangla web documents. Text categorization involves automatically sorting documents into predefined categories. While many methods have been applied to English text categorization, there is limited research on Bangla text categorization. The study aimed to analyze the efficiency of these four methods for categorizing Bangla documents. A Bangla corpus from various websites was developed and used as examples for the experiment. The empirical results showed that all four methods achieve satisfactory performance, with SVM performing particularly well with high-dimensional and relatively noisy document feature vectors [3]. Furthermore, one study has been done recently on classifying Bangla text documents using the latest transformer or attention mechanism-based models. Specifically, the BERT (Bidirectional Encoder Representations from Transformers) and ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) models are applied for Bangla text classification. Both models are pre-trained text encoders, and a fine-tuning approach is used for the downstream classification task. The study utilizes three different Bangla text datasets, with both models demonstrating outstanding performance on two out of the three datasets [4]. Another paper proposed a new method for classifying Bangla news documents using a Deep Recurrent Neural Network (RNN). The collected news data is first preprocessed, and then the model architecture is designed to fit the training data. The model's performance is evaluated using accuracy and F1-score on a testing dataset. The Deep RNN with BiLSTM achieves 98.33% accuracy, outperforming other well-known classification algorithms in Bangla text classification [5]. Sentiment analysis is also a crucial part of NLP task. There is also much research going on sentiment analysis to understand people's emotions expressed on social media and various online platforms in Bangla. One study focused on extracting emotions from Bengali text using Word2vec, Skip-Gram, and Continuous Bag of Words (CBOW) models, with a new Word to Index model emphasizing three distinct emotional classes: happy, angry, and excited. The authors achieved the highest accuracy of 75% using the skip-gram model for classifying these emotions. This study surpasses other existing works using LSTM and CNN models with existing datasets [6].

Overall, all these studies are showcasing the ongoing efforts to advance the field and address the unique challenges posed by Bangla text processing. In our study we will classify Bangla text comments using different machine learning approaches.

Proposed Idea

In recent years, the exponential growth of online content in the Bengali language has led to an increasing need for effective text classification techniques. One area where this is particularly important is in classifying comments on online platforms, which can provide valuable insights into user sentiment and opinions. However, despite the availability of publicly available Bengali text data, there is a lack of comprehensive studies on comment classification using machine learning approaches.

This study aims to fill this gap by proposing a method for classifying Bengali text comments using various machine learning approaches. The goal is to explore the effectiveness of different algorithms, such as Decision Trees, Support Vector Machines, Logistic Regression, Multinomial Naïve Bayes and Random Forest, in accurately categorizing comments into different classes such as not bully, troll, threat, religious and sexual. Our proposed idea includes following key components:

Data Cleaning: Particularly before doing analysis or using the data to make decisions, data cleansing is a crucial stage in the data preparation process. Ensuring the accuracy, reliability, and consistency of data is crucial for deriving useful and valid insights from it. Data cleaning methods frequently involve eliminating duplicates, fixing typos, adding missing numbers, and standardizing data formats. The data cleansing method we employed for our project is listed below.

Removing Null Values: To guarantee that the dataset is free of missing or incomplete data, which might impair the quality of analysis or modeling, removing null values is an essential step in the data cleaning process. We can strengthen the dataset's integrity and prevent problems that can result from missing data by eliminating null rows.

We started our project by finding and eliminating null rows from our text data. In order to remove missing values from the dataset, a methodical check of each row for missing values was performed. We made sure our text data was complete and prepared for additional processing and analysis by doing this.

Removing Duplicate Values: Another important stage in data cleaning is eliminating duplicates to make sure the dataset is clear of unnecessary or duplicated information. By doing this, bias in analysis or modeling is prevented and the dataset's integrity is preserved.

We found and eliminated duplicate rows from our text data for our project. This required methodically looking for duplicates in each row and eliminating them from the dataset. By doing this, we made sure that our dataset only included distinct and relevant data, which raised the standard of our research.

Removing Unnecessary Columns: A critical stage in data cleansing is eliminating irrelevant columns, particularly when focusing on a particular goal like comment classification. In our project, where the goal is to classify comments into five classes, including religious, troll, threat, not bully, and sexual, it is reasonable to remove any columns that are not relevant with the goal of the project. This step makes the dataset simpler and lowers the computational complexity. In general, eliminating irrelevant columns is an essential step in getting dataset ready for classification and guaranteeing the precision and potency of a model.

Removing Punctuation: Punctuation cleansing is an important step in text preparation that improves the readability and uniformity of the text data. To ensure that our content is understandable, we employed a number of punctuations cleansing techniques. In order to make the dataset simple to use for classification tasks, we attempted to eliminate all forms of punctuation, including question marks, exclamation points, commas, and periods. We utilized a method to remove Bangla stop marks from the data because we are working with Bangla.

Removing Noise: In the realm of Natural Language Processing (NLP), noise removal is the art of sifting through text data to remove any elements that could obscure or disrupt the analysis process. This typically involves eliminating irrelevant characters, such as special symbols, numbers, or non-textual elements, that do not contribute to the meaning of the text. In the context of our Bangla dataset, we took several steps to ensure the cleanliness and relevance of our text.

One crucial step was removing Bangla numbers, as they can introduce unnecessary complexity and noise. Additionally, since English is widely used alongside Bangla, especially in online comments, we eliminated all English lines from our text to focus solely on Bangla content. Another common distraction in text data is URLs, which we removed to maintain the focus on the textual content.

Emojis are prevalent in modern communication and can add richness to text, but in the context of our analysis, they were considered noise. Therefore, we employed techniques to remove a wide range of emoji patterns from our data. Despite their popularity, emojis were deemed extraneous to our analysis goals.

Furthermore, we noticed instances of Arabic lines in the comments, which were not relevant to our Bangla-focused analysis. Thus, we also removed Arabic lines from our text data to ensure its purity and relevance to our study. These meticulous steps in noise removal were essential to prepare our dataset for accurate and meaningful analysis in the realm of NLP.

Removing Stop Words: Removing stop words is a common preprocessing step in NLP to filter out common words that do not carry much meaning, such as "the," "is," "and," etc. For English, there are predefined lists of stop words, but for languages like Bangla, these lists need to be collected from external sources.

In our project, we collected a publicly available stop word list from Kaggle for the Bangla language. By using this list, we were able to effectively remove stop words from our dataset, which can help improve the quality of our analysis by focusing on more meaningful words in the text. This shows a proactive approach to data preprocessing, ensuring that our text data is clean and ready for further analysis.

Stemming: Performing stemming in Bangla text is crucial due to the language's morphological complexity, which includes various inflectional and derivational forms of words. Stemming helps to reduce words to their base or root form, which can improve the efficiency and accuracy of text analysis tasks.

In our project, we used the Bangla Stemmer library, which provides a lightweight solution for determining the stem of words in Bangla text. By installing and using this library, we were able to perform stemming effectively, ensuring that our text data was normalized and ready for further processing and analysis.

TD-IDF Vectorizer: Using TF-IDF (Term Frequency-Inverse Document Frequency) Vectorizer is a common practice in text processing tasks, including in Bangla text analysis. TF-IDF is a statistical measure used to evaluate the importance of a word in a document relative to a collection of documents (corpus). It is based on the idea that words that are frequent in a document but rare in the overall corpus are likely to be more important for that document.

In our project, we performed TF-IDF Vectorization on our Bangla text data. This process involved converting the text into numerical vectors based on the TF-IDF scores of each word in the text. By doing this, we were able to represent our text data in a format that machine learning models can understand and use for analysis and classification tasks. TF-IDF Vectorization is a powerful tool for text analysis and can help improve the accuracy and effectiveness of our NLP models.

Algorithms: We employed a Varity of algorithms in our text classification task. The algorithms we employed is giving below:

Multinomial Naïve Bayes: A variation of the Naive Bayes method that is frequently used for text classification applications, including NLP, is called Multinomial Naive Bayes. It performs very well in classification tasks when the features indicate the frequency distribution of each feature, in this case the words or tokens in the text.

Multinomial Naive Bayes is a text classification algorithm that determines a document's likelihood of falling into a specific class based on the frequency of each word in the document. It makes the simplifying premise that the features (word frequencies) are independent of one another, which frequently holds true in practice, particularly for longer documents.

Due to its efficiency and simplicity Multinomial Naive Bayes is a strong algorithm for text classification, making it ideal for large datasets. It performs well with high-dimensional data like word frequencies and is often used as a baseline for comparison in text classification tasks that's we used multinomial naïve bayes in our project.

Logistic Regression: Another popular algorithm in classification task is Logistic Regression. For our multiclass classification in Bangla, logistic regression involve training a model to predict the probability of each class (e.g., religious, troll, threat, not bully, sexual) based on the input features (e.g., words or tokens in the text). The model then predicts the class with the highest probability as the final output.

Logistic regression is suitable for this task because it is relatively simple and interpretable, making it easier to understand the model's predictions.

Decision Tree: Decision Trees are a popular and powerful algorithm for classification tasks due to their simplicity, interpretability, and ability to handle both numerical and categorical data. At each node of the tree, the algorithm makes decisions based on the values of features, splitting the data into smaller subsets that are more homogeneous with respect to the target variable. This process continues recursively, resulting in a tree structure where each leaf node represents a class label. One of the key advantages of Decision Trees is their interpretability; the rules learned by the model can be easily understood and visualized, making them useful for gaining insights into the data. Decision Trees can also handle non-linear relationships between features and the target variable, making them suitable for complex classification tasks.

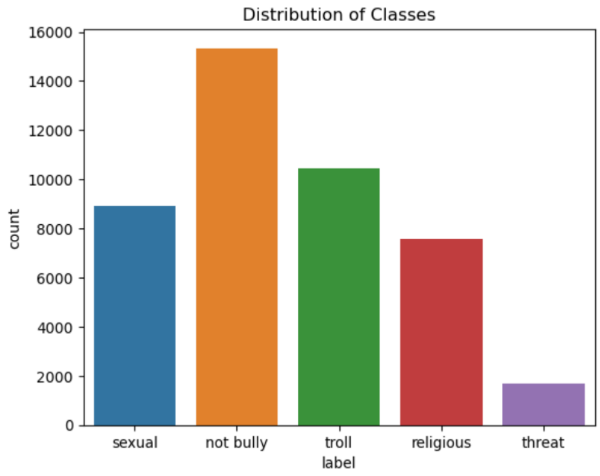
Support Vector Machine: Support Vector Machine (SVM) is a versatile and effective algorithm for classification tasks. SVM works by finding the optimal hyperplane that separates different classes in the feature space while maximizing the margin between the classes. This margin maximization leads to a more robust and generalizable model, making SVM less prone to overfitting, especially in high-dimensional spaces. Additionally, SVM can handle non-linear decision boundaries through the use of kernel tricks, which allows it to learn complex patterns in the data without explicitly mapping the data into a higher-dimensional space. SVM is also effective in small to medium-sized datasets, particularly when the number of features is larger than the number of samples. Moreover, SVM is versatile and can be used for both binary and multi-class classification tasks, as well as for regression and outlier detection. Overall, SVM is a powerful algorithm for classification.

Random Forest: Random Forest is a popular ensemble learning method used for classification tasks in machine learning. It works by constructing a multitude of decision trees during training and outputs the mode of the classes for classification or the average prediction for regression. Random Forest is known for its high accuracy and robustness, making it suitable for a wide range of classification problems. One of the key advantages of Random Forest is its ability to handle large datasets with high dimensionality. It can effectively deal with missing values and maintain accuracy even when a large proportion of the data is missing. Additionally, Random Forest is resistant to overfitting, as each tree in the forest is trained on a subset of the data and features. This helps in reducing variance and improving the overall performance of the model. Random Forest also provides a feature importance score, which can be used to identify the most important features in the dataset. Overall, Random Forest is a powerful and versatile algorithm that is well-suited for classification tasks, especially when dealing with complex and high-dimensional data.

Technical Details

The goal of our project is to explore and implement different machine learning algorithms to classify Bangla text comments into predefined categories. We aim to compare the performance of these algorithms and identify the most effective approach for Bangla text classification. Now we will give a comprehensive overview of the steps we have taken in our project:

Dataset: For our project, we sourced the dataset from "https://data.mendeley.com/datasets/9xjx8twk8p/1". The dataset comprises a total of 44,001 comments, which have been categorized into five classes: not bully, religious, threat, troll, and sexual. This dataset serves as the foundation for our text classification project, providing us with a diverse range of comments to analyze and classify. The dataset's composition allows us to train and test our machine learning models effectively, aiming to accurately classify Bangla text comments into their respective categories. Visualization of class distribution is given below:



Data Cleaning: We used a whole variety of procedures to clean our data.

- **Null Values Removal**: Initially, the dataset contained 44001 comments. After removing null values, the dataset was reduced to 43998 comment.

- **Duplicate Values Removal**: Subsequently, duplicate values were removed from the dataset, revealing a total of 434 duplicate comments. Consequently, the dataset was refined to 43564 unique comments.

- **Removing Unnecessary Columns**: Our dataset contains following columns:
  1. Comment
  2. Category
  3. Gender
  4. Comment react number
  5. Label

Since we want to classify comment with desired label that’s why remove other columns excluding Comment and Label.

- **Removing Punctuation:** We removed a whole range of punctuation marks from our data including Bangla punctuation marks also.

- **Removing Bangla Number:** Since we are working with Bangla text comments there is high chance there will be Bangla number that’s why we employed this technique.

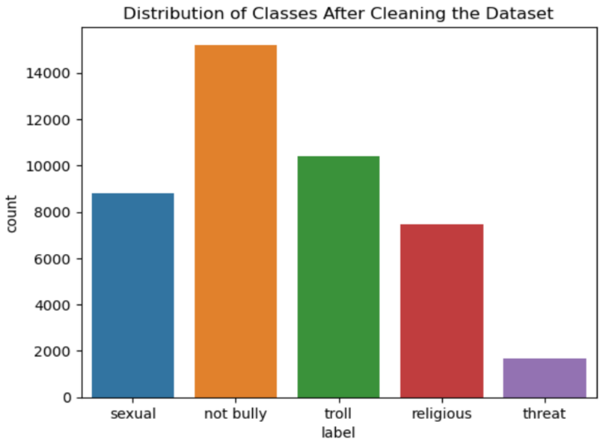
- **Removing English text and Number:** English is a widely popular language. People use this language while writing and talking in their own language. Since we are working on solely Bangla text that’s why we removed English text from our dataset.

- **Removing URL Pattern:** We employed a technique to remove URL pattern from our data.

- **Emoji Removal:** We all know people use all kinds of emoji patterns while writing comments in social media. Since we are working with banal text comments, we employed a whole variety of techniques to remove emoji from text.

- **Arabic Text Remove:** By observing our data we found out Arabic text is also present in our dataset that’s why we employed this technique to remove Arabic text from data.

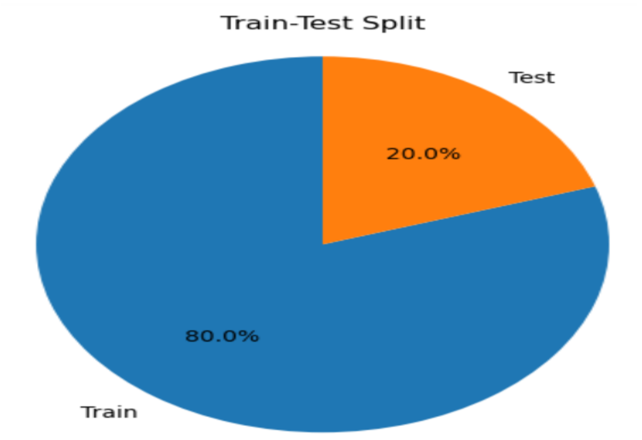
By applying all these techniques, we cleaned the raw data and made it suitable for our classification task. During this process we lose some data also. The visualization of new cleaned distribution of classes is given below:



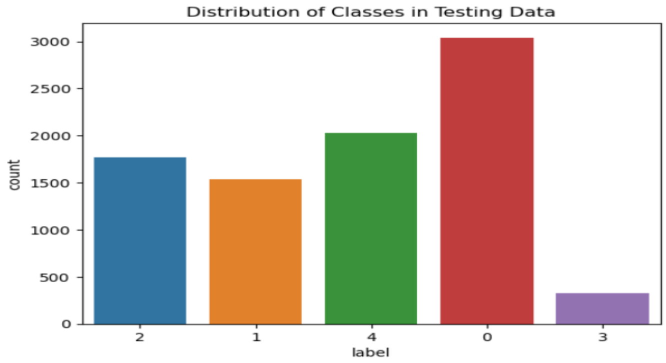
Removing Stop Words: For English, predefined stop word lists are available, but for languages like Bangla, these lists need to be collected from external sources. In this project, we used a publicly available stop word list from Kaggle on Bangla to remove stop words.

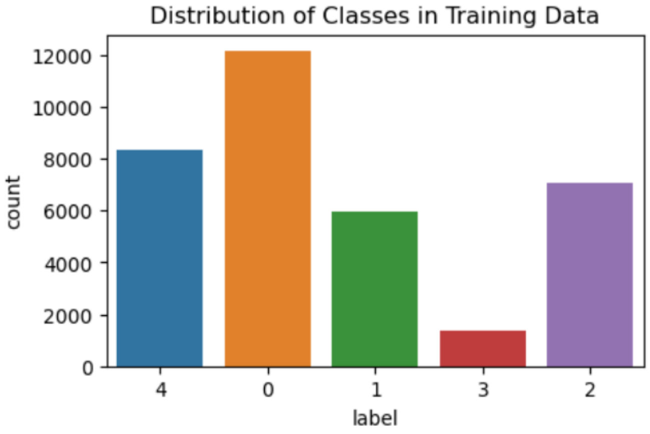
Steaming: Stemming in Bangla text is essential due to its morphological complexity, which includes various word forms. The Bangla Stemmer library was used in the project to reduce words to their base form, improving efficiency and accuracy in text analysis. This ensured that the text data was normalized and ready for processing.

Splitting the Dataset: We split the data dataset in a way where training size has 80% of data and test size has 20% of data.



The visualization of class distribution in training set and testing are given below:





TF-IDF Vectorizer: It is a very common algorithm to extract features from text and turn texts into a meaningful vector representation. We can't feed text to machine learning algorithms, that's why it is a very important step. We used TF-IDF vectorizer to convert text into numerical representation.

System architecture of our model is given below:

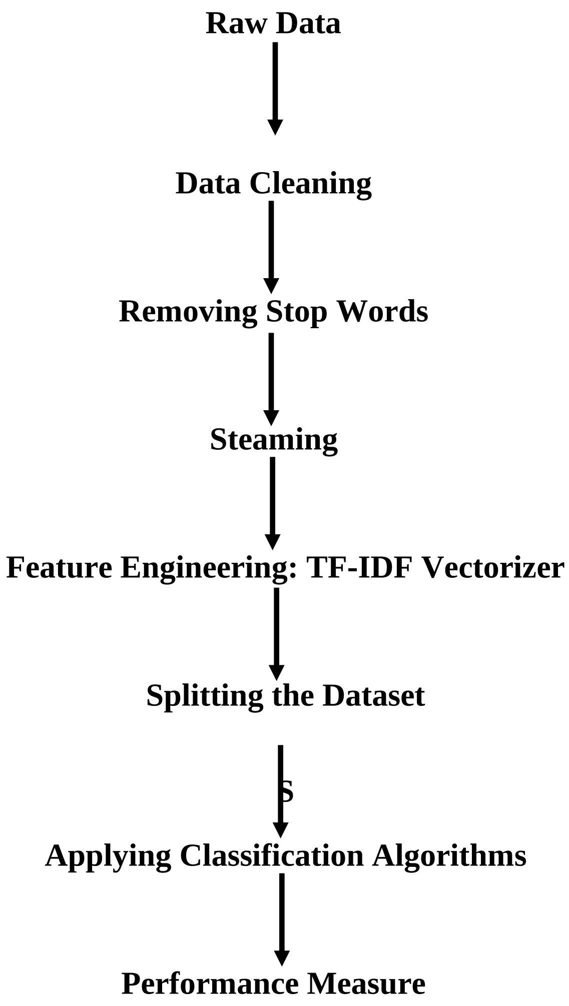
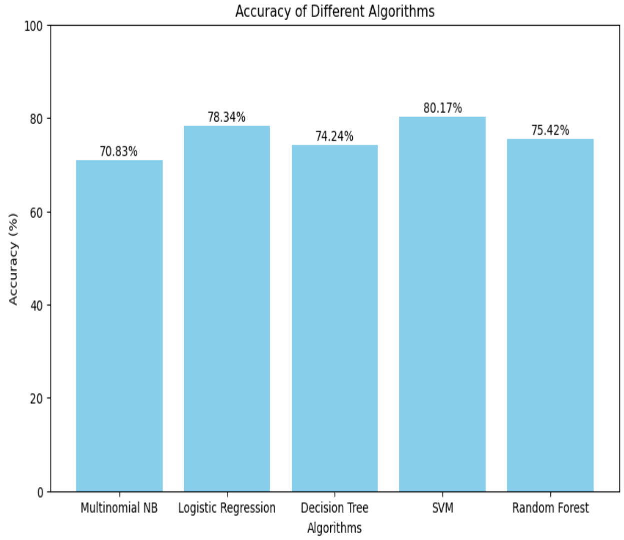


Figure 1: System Architecture of Our Project

Result Analysis

After training our algorithms we applied the text data to the algorithms to see how perfectly models can classify the text. Logistic Regression achieved an accuracy of 78.34%, making it a solid performer. SVM outperformed the other algorithms with the highest accuracy of 80.17%, showcasing its effectiveness in handling complex classification tasks. Decision Tree and Random Forest yielded accuracies of

74.24% and 75.42% respectively, demonstrating their suitability for this task. Multinomial Naïve Bayes, while still performing reasonably well with an accuracy of 70.83%, fell slightly behind the other algorithms. It's evident that SVM is the most suitable algorithm for our Bangla text comment classification task.



Conclusion

In conclusion, our research focused on the classification of Bangla text comments into five categories: not bully, sexual, religious, threat, and troll. We employed various machine learning algorithms, including Logistic Regression, SVM, Decision Tree, Random Forest, and Multinomial Naïve Bayes, to classify the comments based on their content.

Our results indicate that SVM outperformed the other algorithms with an accuracy of 80.17%, followed by Logistic Regression with an accuracy of 78.34%. Decision Tree and Random Forest achieved accuracies of 74.24% and 75.42% respectively, while Multinomial Naïve Bayes achieved an accuracy of 70.83%. These findings suggest that SVM is the most suitable algorithm for our Bangla text comment classification task.

Overall, our research demonstrates the effectiveness of machine learning algorithms in classifying Bangla text comments, which can have significant implications for online content moderation and sentiment analysis in Bangla-speaking communities. Further research could explore additional features or algorithms to improve classification accuracy and expand the application of these techniques to other languages and text classification tasks.

References

[1]. Mirończuk, M. M., & Protasiewicz, J. (2018). A recent overview of the state-of-the-art elements of text classification. *Expert Systems with Applications*, *106*, 36-54.

[2]. Salehin, K., Alam, M. K., Nabi, M. A., Ahmed, F., & Ashraf, F. B. (2021, December). A comparative study of different text classification approaches for bangla news classification. In *2021 24th International Conference on Computer and Information Technology (ICCIT)* (pp. 1-6). IEEE.

[3]. Mandal, A. K., & Sen, R. (2014). Supervised learning methods for bangla web document categorization. *arXiv preprint arXiv:1410.2045*.

[4]. Rahman, M. M., Pramanik, M. A., Sadik, R., Roy, M., & Chakraborty, P. (2020, December). Bangla documents classification using transformer based deep learning models. In *2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI)* (pp. 1-5). IEEE.

[5]. Rahman, S., & Chakraborty, P. (2021, May). Bangla document classification using deep recurrent neural network with BiLSTM. In *Proceedings of International Conference on Machine Intelligence and Data Science Applications: MIDAS 2020* (pp. 507-519).

[6]. Rahman, M., Talukder, M. R. A., Setu, L. A., & Das, A. K. (2022). A dynamic strategy for classifying sentiment from Bengali text by utilizing Word2vector model. *Journal of Information Technology Research (JITR)*, *15*(1), 1-17.
Singapore: Springer Singapore.