Comparing the accuracy of text classifcation using multiple stemming and machine learning method

1st Mithila Arman
Department of Computer Science And Engineering
Brac University
mithila.arman@g.bracu.ac.bd

2nd Md Humaion Kabir Mehedi
Department of Computer Science And Engineering
Brac University
humaion.kabir.mehedi@g.bracu.ac.bd

3rd Mehnaz Ara Fazal
Department of Computer Science And Engineering
Brac University
mehnaz.ara.fazal@g.bracu.ac.bd

4th Annajiat Alim Rasel
Department of Computer Science And Engineering
Brac University
annajiat@bracu.ac.bd

***Abstract*—This paper presents a text classification framework leveraging natural language processing (NLP) techniques and machine learning models. The study focuses on the 20 Newsgroups dataset, employing a preprocessing pipeline that involves tokenization, stemming, and stopword removal. Feature extraction is achieved through the Term Frequency-Inverse Document Frequency (TF-IDF) representation. Two classifiers, Naive Bayes and Support Vector Machines (SVM), are trained and evaluated on the dataset, showcasing their effectiveness in categorizing documents. The classifiers are assessed based on accuracy and detailed classification reports. Results indicate promising performance, demonstrating the proposed methodology's applicability to document classification tasks.**

***Index Terms*—Text Classification, Natural Language Processing (NLP), Term Frequency-Inverse Document Frequency (TF-IDF), 20 Newsgroups Dataset, Stemming, Stopword Removal, Support Vector Machines (SVM), Naive Bayes Classifier, Feature Extraction, PCA, n-gram**

I. Introduction

In the ever-expanding landscape of information, efficient organization and categorization of textual data play a pivotal role in facilitating effective information retrieval and analysis. This paper introduces a comprehensive approach to text classification, a fundamental task in natural language processing (NLP) and machine learning. Our focus lies on the widely recognized 20 Newsgroups dataset, a diverse collection of documents sourced from various newsgroups, presenting a challenging testbed for classification algorithms.

Motivated by the need for automated methods to sift through vast amounts of textual information, we propose a robust preprocessing pipeline. This pipeline incorporates essential techniques such as tokenization, stemming, and stopword removal to transform raw text into a format conducive to machine learning analysis. The subsequent feature extraction phase employs the Term Frequency-Inverse Document Frequency (TF-IDF) representation, capturing the importance of words in the context of the entire document collection.

Two prominent classifiers, Naive Bayes and Support Vector Machines (SVM), are employed to categorize documents based on the derived features. Our investigation aims to assess the efficacy of these classifiers in handling the inherent complexities and nuances present in real-world textual data.

The significance of this work lies in its contribution to the broader field of text classification, shedding light on the performance of well-established techniques on a benchmark dataset. Through rigorous evaluation metrics, including accuracy and detailed classification reports, we aim to provide insights that will aid researchers and practitioners in choosing appropriate methods for similar tasks.

II. Literature review

The paper "Improving the accuracy of text classifcation using stemming method, a case of non-formal Indonesian conversation" emphasizes the importance of Python programming and the Natural Language Toolkit (NLTK) for natural language processing tasks, including preprocessing activities like case folding, tokenizing, and stop word removal [1].

The study used the Communications of the Association for Computing Machinery (CACM) collection, which contains 3204 documents, 64 queries, and relevance judgments. [2] The authors also mentioned various stemming algorithms, such as the PaiceHusk stemmer, Porter's stemmer, and Lovin's stemmer, which have been developed to reduce words to their root forms. Lemmatization has been used in several languages for information retrieval and has shown to improve retrieval performance.Mean Average Precision (MAP) was used to evaluate document relevancy, which combines both recall-oriented and precision-oriented aspects of the search engine.

The paper focuses on Turkish text classification and reviews the available literature on this topic. [3]The paper analyzes the performance of various algorithms in classifying customer inquiries received by an institution. The findings of this study

and the text classification technique utilized can be applied to data in dialects other than Turkish.

The paper highlights that many marketing research studies rely on SVM or LIWC without providing a rationale for their method choice, suggesting a lack of guidance from computer science research on text classification performance. The authors note that lexicon-based approaches, particularly LIWC, perform poorly compared to machine learning methods, with accuracies sometimes only slightly exceeding chance. They suggest that marketing research should consider alternatives such as NB and RF. [4]

The paper mentions two specific statistical stemmers: the N-Gram Stemmer and the HMM Stemmer. The N-Gram Stemmer uses a string-similarity approach and is language independent but requires significant memory and storage. The HMM Stemmer is based on the concept of Hidden Markov Models. [5] The paper also briefly mentions inflectional and derivational methods in stemming, which involve analyzing both inflectional and derivational morphology. The Krovetz Stemmer is mentioned as an example of a linguistic lexical stemmer.

Leopold Kinermann (2002) studied different weight schemes for text representation in input space using SVM . Lin et al. (2006) used SVM for question classification in Chinese . [6] Nguyen Luong (2006) achieved a classification accuracy of 80.72% for Vietnamese text classification using SVM. Pham Ta (2017) achieved a classification accuracy of 99.75% for Vietnamese text classification using a neural network method

Stemming has various applications such as text classification, text clustering, sentiment analysis, text compression, question answering, text summarization, machine translation, and detecting wicked websites. [7]The authors conducted experiments using Urdu language and concluded that existing evaluation metrics can only measure an average conflation of words and cannot perfectly measure stemmer performance for all languages.

This paper highlights the advantages of NLP, such as automatic summary generation, co-reference resolution, and improved efficiency in document retrieval.The paper also mentions the disadvantages of NLP, including the loss of visual context, difficulties in generalized searches, and challenges with synonyms. [8] It emphasizes the importance of deep linguistic understanding in lemmatization for accurate outcomes, while stemming simply chops off word endings without considering meaningful information.

Many machine learning algorithms have been applied to create automatic text classifiers, trained on classified training documents. Support Vector Machine (SVM), k-Nearest Neighbor (k-NN), Logistic Regression (LR), Multinomial Naive Bayes (MNB), and Random Forest (RF) are machine learning algorithms used in this work for text classification. The efficiency of these algorithms on different datasets is analyzed and compared, with Logistic Regression and Support Vector Machine outperforming other models on the IMDB dataset, and k-NN outperforming other models on the SPAM dataset. [9]

[10]Previous studies have shown that no stemming algorithm has utilized the science of phonology, which motivated the authors to explore the use of phonetic algorithms for conflation. Some relevant papers mentioned here also: Krovetz (1993) discusses morphology as an inference process. Lovins (1968) presents the development of a stemming algorithm. Mayfield and McNamee (2003) propose a single N-gram stemming approach. Majumder et al. (2007) introduce YASS, a suffix stripper for stemming. Singh (2003) discusses search algorithms. Tamah (2008) presents work on error-free stemming. UzZaman and Khan (2005) propose T12, a text input system with phonetic support. Xu and Croft (1998) discuss corpus-based stemming using co-occurrence of word variants

III. Dataset

The 20 Newsgroups dataset, a widely utilized benchmark in the field of natural language processing (NLP) and machine learning, stands as a testament to its enduring significance in research and experimentation. Comprising a diverse collection of approximately 20,000 newsgroup documents, the dataset spans an array of topics, ranging from politics and sports to technology and religion. Each document is associated with one of the 20 distinct newsgroups, reflecting a rich tapestry of human discourse across various domains.

This meticulously curated dataset was originally compiled for the purpose of fostering the development and evaluation of text classification algorithms. The documents within each newsgroup exhibit genuine, unfiltered discussions, providing a realistic and challenging environment for the assessment of classification models. Moreover, the dataset's longevity and continued use in research attest to its enduring relevance as a standard benchmark for text classification tasks.

Noteworthy features of the 20 Newsgroups dataset include its representation of real-world challenges, such as document length variability, diverse writing styles, and the presence of noise in the form of headers, footers, and quotes. Addressing these challenges becomes paramount for researchers seeking to develop robust and adaptable text classification algorithms.

In conclusion, the 20 Newsgroups dataset remains a cornerstone in the field of text classification research, offering a curated and diverse collection of documents that mirrors the complexity of real-world textual data. Its enduring popularity stems from its ability to simulate authentic challenges, providing a standardized platform for the evaluation and comparison of text classification methodologies. As a foundational resource, the 20 Newsgroups dataset continues to inspire advancements in NLP and machine learning, fostering innovation and contributing to the collective understanding of effective strategies for handling diverse and intricate textual information.

IV. Methodology

In this study, we present a comprehensive methodology for text classification applied to the 20 Newsgroups dataset. The initial step involves the collection of documents from diverse newsgroups, with subsequent preprocessing to remove metadata such as headers, footers, and quotes. Leveraging the NLTK library, tokenization breaks down the documents into individual words, setting the stage for subsequent analysis.

A crucial aspect of our text normalization process involves converting all text to lowercase and removing special characters, numbers, and punctuation. This ensures uniformity and consistency in word representation across the dataset. To further enhance the efficiency of our analysis, stemming is applied using the Porter Stemmer, reducing words to their base or root form.

Stopword removal, the next step in our methodology, targets common words with minimal semantic meaning, contributing to the refinement of the dataset. Part-of-speech tagging is employed to identify the grammatical category of each word, with a specific focus on retaining nouns (NN) and verbs (VB) to preserve words with significant semantic content.

The heart of our feature extraction lies in the utilization of the Term Frequency-Inverse Document Frequency (TF-IDF) representation. This technique transforms preprocessed and filtered tokens into numerical features, capturing the importance of words within the broader context of the entire document collection.

For model training and evaluation, we employ the train-test-split function to partition the dataset into training and testing sets. Two classification models, Multinomial Naive Bayes and Support Vector Machines (SVM), are chosen for their effectiveness with text data. These models are trained on the TF-IDF features extracted from the training data and subsequently evaluated on the testing set.

Performance evaluation is a critical component of our methodology, employing standard metrics such as accuracy and a detailed classification report. The latter provides a granular understanding of precision, recall, and F1-score across different categories, offering insights into the strengths and limitations of the chosen classifiers.

The results of our study are subjected to thorough analysis, contributing valuable insights into the capabilities and nuances of the proposed methodology. This comprehensive approach to text classification, encompassing preprocessing, feature extraction, model training, and evaluation, adds to the collective understanding of effective methodologies in handling the complexities of real-world textual data.

References

[1] E. P. W. Rianto1*, Achmad Benny Mutiara2 and P. I. Santosa, "Improving the accuracy of text classifcation using stemming method, a case of non-formal indonesian conversation," *Springer*, p. 16, 2021.

[2] V. Balakrishnan and E. Lloyd-Yemoh, "Stemming and lemmatization: A comparison of retrieval performances," *semantic scholar*, vol. 2, no. 3, p. 6, 2014.

[3] . Yehia Ibrahim Alzoubi 1, Ahmet E. Topcu 2 and A. E. E. 3, "Machine learning-based text classification comparison: Turkish language context," *Applied Science, MDPI*, no. 2, p. 22, 2023.

[4] C. S. M. H. Jochen Hartmann∗, Juliana Huppertz, "Comparing automated text classification methods," *ELSEVIER*, October 2018.

[5] M. G. Jivani, "A comparative study of stemming algorithms," *ISSN*, vol. 2(6), p. 1938, 2011.

[6] L. T. M. Nguyena, "Text classification based on support vector machine," *Mathematics*, vol. 9, no. 2, p. 19, 2019.

[7] M. I. T. S. H. A. A. Abdul Jabbar1, Sajid Iqbal2, "Empirical evaluation and study of text stemming algorithms," *Springer Nature*, 2020.

[8] N. N. M. D. B. M. . B. I. o. T. . . . Divya Khyani1, Siddhartha B S2, "An interpretation of lemmatization and stemming in natural language processing," *ISSN*, vol. 22, January 2021.

[9] K. A. Sayar Ul Hassana, Jameel Ahameda, "Analytics of machine learning-based algorithms for text classification," *sciencedirect*, vol. 3, pp. 238–248, 2022.

[10] P. H. S. D. B. P. Pande, "Application of natural language processing tools in stemming," *International Journal of Computer Applications (0975 – 8887)*, vol. 27, August 2011.