International Journal of Interdisciplinary Innovative Research & Development (IJIIRD)
ISSN: 2456-236X
Vol. 10 Issue 01 | 2025

Comparative Analysis of Machine Learning Algorithms for Healthcare Text Classification

Mrs. Rane Seema Vijay[1], Dr. Vinod M. Patil[2]

[1] Assistant Professor, Department of Computer Science, Smt. G. G. Khadse College, Muktainagar, Dist.: Jalgaon, Maharashtra, India
[2] Professor & Head Department of Computer Science, Shri Shivaji College, Akola India, Maharashtra, India

DOI: 10.5281/zenodo.17666902

ABSTRACT

In the current context, there is a substantial amount of online data are available across various topics on the internet, leading to a rapid growth in textual data. Consequently, it becomes crucial to efficiently organize this data to facilitate easy retrieval of important data and to prevent data loss. One effective approach to address this challenge involves categorizing the data into distinct classes or extracting the most pertinent and valuable information. This research paper endeavors to tackle this issue by classifying healthcare texts into different categories. To achieve this objective, three distinct machine learning algorithms, Support Vector Machines (SVM), Decision Tree Classifier (DTC), Random Forest (RF), Logistic Regression (LR) and Multinomial Naive Bayes (MNB) and Ensemble learning, are employed. An experiment is conducted using a dataset containing different healthcare related text collected from various websites.

Keyword: - healthcare text, machine learning, SVM, Ensemble learning, n-Gram

1. INTRODUCTION

Text classification techniques primarily serve the purpose of determining the categorization of articles based on particular topics into their predefined categories. The objective of classification is to ascertain whether a given text aligns with a specific category. The continuous evolution of information technology directly contributes to the escalation of data volume. An effective approach for distilling essential and noteworthy information from this vast pool of data involves categorizing it into distinct classes or categories. The rapid expansion in data volume amplifies the intricacy of classification, demanding significant time and effort for manual categorization. These factors, among others, underscore the necessity and significance of automating the classification of digital data.

Any text holds significant importance for obtaining timely information that may also be needed in the future. In today's era, the internet is saturated with various health related content, posing a formidable challenge in efficiently storing this data for seamless retrieval in the future. All the articles often contain substantial irrelevant information that holds no value to the reader, with the user seeking concise points about the data. To address this issue, the concept of health text classification is employed, wherein text is categorized into different classes in health domain texts categories in physical, nutritional, spiritual, social, emotional, environmental and intellectual health texts. Consequently, this paper delves into the task of automating the classification of healthcare text data.

A large number of scientific studies on English text classification have been performed on 20 newsgroups, reuters--21578, IMDb, Twitter, Amazon etc. datasets. There are studies in this area for other languages using for the automated classification of Uzbek news articles with the use of the word-level n-gram and character level n- gram models with the TF-IDF algorithm when performing the multi-class text classification, using 6 machine learning algorithms Support Vector Machines, Decision Tree Classifier, Random Forest, Logistic Regression and Multinomial Na¨ıve Bayes [1].

In this paper, for the automated classification of healthcare texts dataset used TF- IDF algorithm and TF-IDF using n-gram model. When performing the multi-class text classification, a group of 6 machine learning algorithms is used: Support Vector Machines, Decision Tree Classifier, Random Forest, Logistic Regression, Multinomial Naive Bayes and Ensemble learning.

2. RELATED WORK

In 2021 [2] tested SVM, NB, and LR for three types of datasets from the English news website for text classification approach and named these three categories as Science and Technology. Precision, recall and F1-Score is used as evaluation metrices. Experiment performed on Weka and found that SVM performs better than rest of machine learning techniques used.[3] This research paper focuses on improving text classification by implementing advanced preprocessing techniques. Specifically, ECAS stemmer is utilized to identify root words, Efficient Instance Selection reduces text data dimensionality, and Pre-computed Kernel Support Vector Machine is employed for classification. The study involves experiments with 750 research articles categorized into three classes: engineering, medical, and educational articles.

The EIS-SVM classifier demonstrates superior real-time research article classification performance, achieving a 75% accuracy rate. This suggests that ECAS stemmer and EIS are effective in this context. Future work aims to extend the classifier to handle more than two classes.

In this research [4], the emphasis was placed on evaluating the effectiveness of two feature extraction methods: TF-IDF (Term Frequency-Inverse Document Frequency) and Doc2vec (Document to Vector). These techniques were applied to datasets from Cornell movie reviews, UCI sentiment labeled data, and Stanford movie reviews. The resulting features, derived from the processed text sentences, were then employed for training and testing with a range of classifiers, including Logistic Regression, Support Vector Machines, K-Nearest Neighbors, Decision Tree, and Bernoulli Naive Bayes.

Fang Miao et.al., (2018) [5], in their study on text classification techniques are applied and analyzed with different machine learning classification algorithms. The researcher in this study utilizes the Chinese news text corpus from Fudan University's news corpus. This dataset comprises news content spanning nine distinct categories, with the entire corpus being divided into training and testing sets. The author applies a four-step classification model sequentially to each text. In this research, the TF-IDF algorithm is employed for text vectorization. For the classification task, the author selects the Naive Bayes, K-nearest Neighbor, and Support Vector Machine classifiers. Among these, the SVM method exhibits the highest performance, achieving a precision, recall, and F-value of 97%. Additionally, both the Naive Bayesian and K- nearest neighbor techniques yield similar results, scoring 92% across all parameters.

L. Junyeon, et.al., (2016) [6], in this study, the authors gathered 340 articles from The Korea Herald that covered topics related to 'North Korea' and 'nuclear energy.' These articles were published in January 2016. The features were extracted using a term weighing function, and the remaining words were utilized to construct a word-document matrix, which was subsequently converted into a simple frequency matrix. The classification task involved the use of SVM classifiers, k-NN, decision trees, and Bayesian statistical classification. Evaluation and comparison were conducted using standard information classification metrics, including recall rates and F-measures. The findings revealed that the SVM model outperformed the other models, achieving the highest F-measure value of 59%.

In this study [7], they utilized an optimized SVM approach to assess sentiment analysis in the context of Twitter data, gold dataset, and movie reviews. They conducted a comparative analysis between the Optimized Support Vector Machine and the conventional Support Vector Machine and Naive Bayes classifier. Furthermore, they fine-tuned the hyperparameter values of the RBF kernel SVM, resulting in enhanced accuracy rates for the Gold Movie and Twitter datasets, achieving 73.5%, 74.5%, and 78%, respectively. The optimal hyperparameter values were determined through the proposed methodology, which led to more precise dataset classification compared to the current system. The study also noted the versatility of SVM kernels and their associated hyperparameters, offering opportunities for performance improvement.

3. PROPOSED METHODOLOGY

In this section, we describe the steps involved in executing the multi-class text classification for healthcare text data.

3.1 Data Collection

First phase in classification is text collection. For the proposed system text dataset from health domain has been formed which categories into seven dimensions of health i.e. physical, nutritional, spiritual, social, emotional, environmental and intellectual health. Entire collection is divided into training and testing text dataset.

3.2 Pre-Processing

Text preprocessing is a pivotal aspect of text classification, significantly impacting the performance of classifiers. Initially collected data often isn't in a suitable format for direct training and necessitates processing beforehand. Text data can have missing values or noise, and various preprocessing methods have been developed, the choice of which depends on the specific context. This preprocessing phase is essential for eliminating erroneous or invalid data.

Data preprocessing, the second phase, holds particular significance in text mining. It transforms text documents into a format amenable to automatic processing. This involves tasks like tokenization, where punctuation, special characters, and numbers are removed, leaving behind terms or tokens. In this study, we employed a self-created dataset sourced from health domain websites. The primary objective of preprocessing is to represent text data as feature vectors, essentially breaking down text into discrete words. To achieve this, we structured the text dataset as relationships, where the details are phrases or words from the text dataset, along with their corresponding categories.

Many text datasets tend to include superfluous words like stop words, typos, and slang, which can adversely affect the performance of various algorithms, especially probabilistic and statistical learning ones. In this section, we briefly discuss techniques and methods for text cleaning and preprocessing text datasets.

a. Tokenization

Another pre-processing method is Tokenization which breakdowns a text sentences into tokens, which can be words, phrases, symbols, or other significant components [1] the search for words in a sentence is the primary goal of this step [9]. A parser that handles the tokenization of the text data is necessary for both text classification and text mining for example:

Sentence:

Natural language processing (NLP) is a field of computer science concerned with the interactions between computers and human.

For above given sentence, the tokens are as follows:

['Natural', 'language', 'processing', '(', 'NLP', ')', 'is', 'a', 'field', 'of', 'computer', 'science', 'concerned', 'with', 'the', 'interactions', 'between', 'computers', 'and', 'human']

b. Stop Words

Classification of Text contains numerous words which do not contain significant implication in classification algorithms to be used, such as “a”, “about”, “above”, “across”, “act”, “affects”, “after”, “again”, “appear”, Eliminating these words from texts is the method used to deal with them most frequently[10].

c. Noise Removal

The majority of text records include a lot of extraneous characters, like punctuation and special characters. While crucial for human comprehension of text data, significant punctuation and special characters can be harmful to classification organizations [11].

d. Stemming

In NLP, a single word may take on different forms (such as the singular and plural noun forms) even though each form has the same semantic meaning [11]. Stemming is one technique for combining into a single feature space. Text stemming alters words using various linguistic techniques, such as affixation, to produce new word forms (addition of affixes). For example, the stem of the word “programers” is “program”, “languages” is “languag”,

3.3 Feature Extraction

Feature extraction is the process of converting raw data into another type of data that is operated on by the algorithm called feature extraction. Feature extraction creates a new characteristic and reduced feature set that represents most of the useful information in the data [13]. The process of selecting a list of words from the text data and turning them into a feature set that a classifier may use is known as feature extraction of text. The review of available feature extraction techniques was a focus of this work. The following methods can be applied to extract features from text data:

a. **N-Gram**

The n-gram approach uses a set of n words that appear in a text dataset "in that order." Although it is not a text description, this could be used to identify a text. Text categorization and tasks involving natural language processing frequently use n-grams of texts. An n-gram is a grouping of n words in close proximity. Size 1 n-grams are also referred to as "unigrams", size 2 as "bigrams," and size 3 as "trigrams," respectively [8].

N-gram as the arrangement of N words, by that conception, a 2-gram (or bigram) is a sequence of two-word like “please help”, “to turn”, or “around cupboard”, and a 3-gram (or trigram) is a sequence of three-word like “please help to”, or “turn around cupboard”

b. **TF-IDF**

The bag of words method has the drawback that the data tends to favor the words with higher frequency. The model might not learn much from these words. Domain-specific terms with lower scores may be deleted or disregarded as a result of this issue. This issue is fixed by rescaling the frequency of the words based on how frequently they appear throughout all documents. As a result, the scores for words that appear often throughout all documents are lower [3] [4].

- Term Frequency - Inverse Document Frequency is the name of this scoring method.
- The rate at which a word appears in the present document is known as the term frequency (TF).
- The score of the words across all the papers is known as Inverse Document Frequency (IDF).

These ratings might draw attention to the words that are distinctive, i.e., the words that convey important information in a given document. As a result, the IDF of a rare term is high and the IDF of a common term is low. From the above reviewed articles TF-IDF with NB[2], TF-IDF with SVM [1][5][6][7] gives more effective results as compared with other extraction methods.

3.4 Feature Selection

Feature selection is the method of indicating specific features, from features set which helps in shortening the training time as well as regularization. The appropriate feature is important for training purposes since it helps to improve the classification model accuracy. However, the inappropriate features might negatively affect performance [12]. Mostly chi-square filter feature selection method was used to discover the important features from the texts which helps in increasing the performance of the model.

3.5 Classification Models

In the process of conducting multi-class text classification on healthcare-related textual data, we employed six distinct machine learning algorithms: Support Vector Machines, Random Forest, Logistic Regression, K- nearest neighbor and Multinomial Naive Bayes. The model we devised to address this challenge is illustrated in the functional diagram below Figure 1.

Both the using Tf-idf and N-Gram using Tf-idf feature vectors were fed to various commonly used classifiers in pattern recognition problems namely Random Forest (RF), Support Vector Machine(SVM), Naive Bayes Multinomial (NBM), Logistic Regression, K- nearest neighbor and proposed ensemble model. These algorithms are briefly illustrated in the subsequent paragraphs.

Support Vector Machines (SVM):

SVM is a powerful classification algorithm used for both binary and multi-class classification tasks.It works by finding the hyperplane that best separates different classes in the feature space. SVM aims to maximize the margin between classes, making it robust to overfitting. It is effective in high-dimensional spaces and is versatile due to various kernel functions. Support Vector Machines (SVMs) standout as highly resilient and effective classification algorithms utilized in both classification and regression analyses. This advanced classification technique is applicable to both linear and nonlinear datasets, employing a nonlinear mapping process to elevate the original training data to a higher-dimensional space.

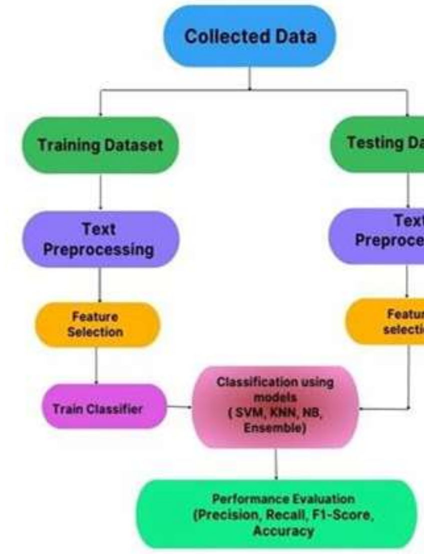

Figure 1: The model structure diagram of the proposed model.

Random Forest:

Random Forest is an ensemble learning method that combines multiple decision trees. Each tree in the forest is constructed using a random subset of the data and features. It's robust, handles high- dimensional data well, and provides feature importance scores. The Random Forest algorithm, a supervised learning method, finds application in both classification and regression tasks, although it is predominantly employed for classification problems. Within the Random Forest, decision trees are constructed from data samples, computations are derived from each of them, and ultimately, the best solution is determined through a voting mechanism.

Logistic Regression:

Logistic Regression is a linear classification algorithm used for binary and multi-class problems. It models the probability of a data point belonging to a particular class. Logistic Regression is interpretable and can provide insights into feature importance. Regularization techniques like L1 and L2 help prevent overfitting. Logistic regression is a predictive algorithm designed for binary outcomes, where the result can be characterized as either positive or negative, such as Yes/No, Presence/Absence, Pass/Fail. Essentially, it signifies whether an event occurs or doesn't occur. This algorithm assesses variables in relation to one another to determine the final outcome, which falls into one of two categories, represented as 0 or 1.

$$P(Y=1|X) \text{ or } P(Y=0|X)$$

The independent variables may encompass both categorical and numeric types, but the dependent variable consistently retains its categorical nature. It represents the likelihood of the dependent variable Y based on the independent variable X.

K-Nearest Neighbor (K-NN):
K-NN is a simple instance-based learning algorithm used for classification and regression. It classifies a data point based on the majority class of its K-nearest neighbors in the feature space. K- NN's performance depends on the choice of K and the distance metric used. It is sensitive to the scale of features and may require feature scaling.
According to Bayes' Theorem, represent a group of classification algorithms. They are not individual algorithms but rather a category of algorithms that all adhere to a common principle: the independence assumption between pairs of features being classified. These classifiers come into play particularly when dealing with high-dimensional input data.

Ensemble learning:
Ensemble learning enhances machine learning outcomes by amalgamating multiple models, leading to superior predictive performance when compared to a solitary model. The fundamental concept involves training a group of classifiers or experts and enabling them to collectively make decisions through a voting mechanism. It is the process of running two or more related but different analytical models and then synthesizing the results into a single score or spread in order to improve the accuracy. In the ensemble-based approach, multiple models are generated and combined to address complex problems. Instead of relying on a single classifier, ensemble methods utilize a blend of multiple classifiers or predictors, all trained to tackle the same problem, resulting in improved outcomes. Ensemble methods can employ either similar base models or a mix of different types of base models. In ensemble-based machine learning, the performance of a model is optimized by aggregating prediction results from chosen weak models [14].

4. RESULT

The results obtained for both the feature extraction methods, specifically TF-IDF and N- Gram using Tf-idf using all the aforementioned classifiers are presented in Table 1 and Table 2. Table 3 shows the accuracy of all the classifiers and table 4 shows the Comparison of using Tf-idf and N-Gram using Tf-idf in terms of Macro Average Precision. Diagrammatic representation of table 3 and table 4 are shown in Figure 2 and Figure 3 respectively.

Table 1: Using Tf-idf

Classifier	Class	Precision	Recall	F1-Score
LR	Emo	0.5	0.63	0.56
	Env	0.85	0.82	0.84
	Intel	0.76	0.83	0.79
	Nut	0.88	0.84	0.86
	Phy	0.71	0.76	0.73
	Soc	0.69	0.69	0.69
	Spir	0.76	0.49	0.6
RF	Emo	0.56	0.64	0.6
	Env	0.83	0.9	0.86
	Intel	0.78	0.86	0.82
	Nut	0.87	0.89	0.88
	Phy	0.77	0.84	0.8
	Soc	0.71	0.7	0.7
	Spir	0.77	0.46	0.58
KNN	Emo	0.75	0.84	0.79
	Env	0.76	0.72	0.74
	Intel	0.47	0.66	0.55
	Nut	0.92	0.82	0.79
	Phy	0.84	0.84	0.84
	Soc	0.69	0.47	0.56
	Spir	0.71	0.66	0.69
NB	Emo	0.49	0.63	0.55
	Env	0.84	0.88	0.86
	Intel	0.79	0.85	0.82
	Nut	0.89	0.88	0.89
	Phy	0.76	0.77	0.76
	Soc	0.76	0.66	0.71
	Spir	0.73	0.52	0.61
SVM	Emo	0.74	0.67	0.7
	Env	0.83	0.93	0.88
	Intel	0.89	0.95	0.92

	Nut	0.8	0.8	0.8
	Phy	0.72	0.88	0.79
	Soc	0.79	0.38	0.51
	Spir	0.81	0.69	0.74
Ensemble Learning	Emo	0.76	0.99	0.79
	Env	1.00	0.65	0.67
	Intel	1.00	0.5	0.38
	Nut	1.00	0.23	0.93
	Phy	1.00	0.86	0.83
	Soc	0.95	0.65	0.72
	Spir	0.87	0.83	0.82

Table 2: N-Gram Using Tf-idf

Classifier	Class	Precision	Recall	F1-Score
LR	Emo	0.53	0.64	0.58
	Env	0.85	0.84	0.84
	Intel	0.77	0.87	0.82
	Nut	0.88	0.86	0.87
	Phy	0.76	0.79	0.77
	Soc	0.75	0.7	0.73
	Spir	0.68	0.5	0.57
RF	Emo	0.54	0.63	0.58
	Env	0.83	0.86	0.84
	Intel	0.79	0.82	0.81
	Nut	0.83	0.88	0.85
	Phy	0.75	0.84	0.8
	Soc	0.72	0.68	0.7
	Spir	0.72	0.45	0.56
KNN	Emo	0.54	0.65	0.59
	Env	0.84	0.83	0.83
	Intel	0.75	0.78	0.77
	Nut	0.87	0.79	0.83
	Phy	0.73	0.78	0.75
	Soc	0.63	0.69	0.66
	Spir	0.7	0.47	0.57
NB	Emo	0.48	0.61	0.54
	Env	0.82	0.86	0.84
	Intel	0.77	0.81	0.79
	Nut	0.86	0.83	0.85
	Phy	0.75	0.74	0.75
	Soc	0.76	0.69	0.72
	Spir	0.65	0.49	0.56
SVM	Emo	0.47	0.66	0.55
	Env	0.84	0.84	0.84
	Intel	0.75	0.84	0.79
	Nut	0.92	0.82	0.86
	Phy	0.76	0.72	0.74
	Soc	0.71	0.66	0.69
	Spir	0.69	0.47	0.56
Ensemble Learning	Emo	0.84	0.75	0.78
	Env	0.88	0.89	0.89
	Intel	0.79	0.9	0.84
	Nut	0.90	0.89	0.9
	Phy	0.85	0.8	0.82
	Soc	0.82	0.87	0.85
	Spir	0.88	0.84	0.86

Table 3: Comparison of using Tf-idf and N-Gram using Tf-idf in terms of Macro Average Precision

Classifier	Accuracy using Tf-idf	Accuracy N-Gram using Tf- idf
LR	72.28	74.10
RF	75.53	73.85
KNN	71.79	71.53
NB	74.21	71.96
SVM	78.53	71.78
Ensemble Learning	83.18	84.83

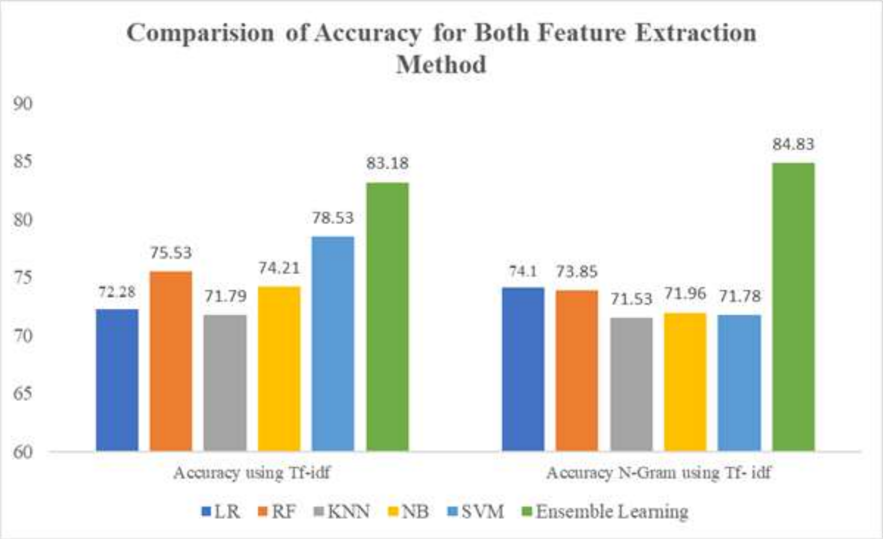

Figure 2: Results of Accuracy for Both Feature Extraction Method

Table 4: Comparison of using Tf-idf and N-Gram using Tf-idf in terms of Macro Average Precision

Classifier	Using Tf-idf	N-Gram using Tf-idf
LR	73	75
RF	76	74
KNN	73	72
NB	75	73
SVM	78	73
Ensemble Learning	84	86

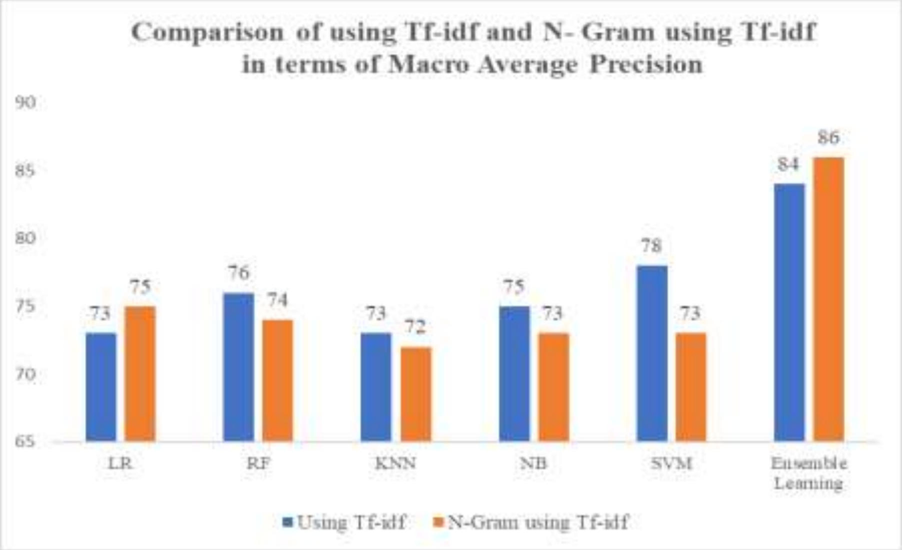

Figure 3: Results of using Tf-idf and N-Gram using Tf-idf in terms of Macro Average Precision

5. CONCLUSION AND FUTURE WORK

The primary aim of this comprehensive analysis is to gain a more profound understanding of the effectiveness of feature extraction methods, specifically TF-IDF and N-Gram using Tf-idf. After evaluating accuracy across dataset, it is evident that both of feature extraction methods exhibit commendable performance in most cases. Notably, Tf-idf tends to outperform N-Gram using TF-IDF in terms of accuracy across the majority of classifiers and gives highest accuracy with proposed ensemble model.

This section presents the performance evaluation of various classifiers in the context of healthcare text classification. This evaluation can be valuable for tasks such as information retrieval and text mining. Among the different classifiers, the highest accuracy was attained using proposed ensemble method achieving an accuracy of 84.83% and average precision of 0.86 when dataset is spits into 60-40 ratio. In the future, we intend to explore preprocessing techniques, feature extraction methods, and feature selection approaches that can handle text data belonging to multiple categories.

6. REFERENCES

[1] I M Rabbimov and S S Kobilov “Multi-Class Text Classification of Uzbek News Articles using Machine Learning” Journal of Physics: Conference Series 1546 (2020) 012097 IOP Publishing doi:10.1088/1742-6596/1546/1/012097

[2] Xiaoyu Luo, “Efficient English text classification using selected Machine Learning Techniques”, Alexandria Engineering Journal (2021) 60, 3401–3409, https://doi.org/10.1016/j.aej.2021.02.009

[3] B. Ramesh and J.G.R. Sathiaseelan, “AN IMPLEMENTATION OF EIS-SVM CLASSIFIER USING RESEARCH ARTICLES FOR TEXT CLASSIFICATION”, ICTACT JOURNAL ON SOFT COMPUTING, ICTACT JOURNAL ON SOFT COMPUTING, APRIL 2016, VOLUME: 06, ISSUE: 03, DOI: 10.21917/ijsc.2016.0170

[4] Avinash M., Sivasankar E. (2019) A Study of Feature Extraction Techniques for Sentiment Analysis. In: Abraham A., Dutta P., Mandal J., Bhattacharya A., Dutta S. (eds) Emerging Technologies in Data Mining and Information Security. Advances in Intelligent Systems and Computing, vol 814. Springer, Singapore. https://doi.org/10.1007/978-981-13-1501- 5_41, PP 475-486.

[5] Fang Miao, Pu Zhang, Libiao Jin, Hongda Wu, “Chinese News Text Classification Based on Machine learning algorithm”, 10th International Conference on Intelligent Human-Machine Systems and Cybernetics, Page No. 48-51, 2018. 978-1-5386-5836-9/18/$31.00 ©2018 IEEE, impact factor: 9.107

[6] L. Junyeon, S. Seungsoo, and K. Jungju, “Comparison of performance in text mining using categorization of unstructured data,” Indian Journal of Science and Technology, vol. 9, no. 24, 2016.

[7] Bhumika M. Jadav, Vimalkumar B. Vaghela, PhD, “ Sentiment Analysis using Support Vector Machine based on Feature Selection and Semantic Analysis”, International Journal of Computer Applications (0975 – 8887) Volume 146 – No.13, July 2016.

[8] Foram P. Shah, Vibha Patel, “A Review on Feature Selection and Feature Extraction for Text Classification” 978-1-4673-9338- 6/16/$31.00 ©2016 IEEE, pp 2264-2268.

[9] Aiman Moldagulova, Rosnafisah Bte. Sulaiman, “Using KNN Algorithm for Classification of Textual Documents”, 8th International Conference on Information Technology (ICIT), Page No. 665-671, 2017, 978-1-5090-6332-1/17/$31.00 ©2017 IEEE.

[10] Foram P. Shah, Vibha Patel, “A Review on Feature Selection and Feature Extraction for Text Classification” 978-1-4673-9338- 6/16/$31.00 ©2016 IEEE, pp 2264-2268.

[11] G. L. Yovellia Londo, D. H. Kartawijaya, H., T. Ivariyani, Y. S. P. W.P., A. P. Muhammad Rafi and D. Ariyandi, "A Study of Text Classification for Indonesian News Article," 2019 International Conference of Artificial Intelligence and Information Technology (ICAIIT), Yogyakarta, Indonesia, 2019,pp.205-208, doi: 10.1109/ICAIIT.2019.8834611, 978-1-5386-8448-1/19/$31.00 ©2019 IEEE

[12] Tehseen Zia, et. al., “Evaluation of Feature Selection Approaches for Urdu Text Categorization”, I.J. Intelligent Systems and Applications, 2015, 06, 33-40 Published Online May 2015 in MECS (http://www.mecs- press.org/) DOI: 10.5815/ijisa.2015.06.03

[13] Foram P. Shah, Vibha Patel, “A Review on Feature Selection and Feature Extraction for Text Classification” 978-1-4673-9338- 6/16/$31.00 ©2016 IEEE, pp 2264-2268.

[14] Deepti Rani, Nasib Singh Gill, Preeti Gulia, Jyotir Moy Chatterjee, "An Ensemble-Based Multiclass Classifier for Intrusion Detection Using Internet of Things", Computational Intelligence and Neuroscience, vol. 2022, Article ID 1668676, 16 pages, 2022. https://doi.org/10.1155/2022/1668676

[15] Tuin S. Experiments on malay short text classification. Proceeding of International Conference on Electrical Engineering and Informatics. 2017: 1-4