Improving Deep Learning Text Classification By Incorporating Vectorization And Agglomerative Data Clustering

Titu Singh Arora[1], Dr. Amit Sharma[2]

[1]Ph.D. Research Scholar, School of Engineering & Technology , Career Point University, Kota, Rajasthan, India

[2]Associate Professor, School of Engineering & Technology , Career Point University, Kota, Rajasthan, India

Abstract *The most prevalent unstructured data in the data world is text data, which has been the subject of several research papers that have been published both online and offline. Users sometimes struggle to determine which machine learning model belongs to which category. An email's text, for instance, can be analysed by a spam detection model to determine whether or not the email is spam. Many machine learning models have been created recently that centre on text classification. This work suggests a research paper provides an innovative strategy of using text clustering as a data pre-processing step to enhance the classification models in order to get around the restrictions. A collection of news stories that have been classified as "genuine" or "fake" serves as the dataset for this study. The two main components of the technique that is being discussed are text data pre-processing and data modelling. First, a sparse matrix is produced by TF-IDF vectorizing the text data from the news articles. Each row in the matrix represents a news story in vectorized form. These vectors are used to perform agglomerative clustering (hierarchical clustering), which adds the functionality of assigning the data to two additional clusters. The accuracy scores are compared after the model has been trained using both the base dataset and the updated dataset in turn in the second phase. It is discovered that the changed dataset offers better outcomes for categorising the bogus news.*

Keywords: Machine Learning, Data Mining,K-means clustering, Spam detection, False positive, Deep Learning, TF-IDF, Text Classification, SVM, Feature Extraction, Document Classification.

I Introduction

In today's society, having a social presence in a community and an online media presence are both necessary to be regarded as linked to others. People have discovered a variety of novel methods to engage with one another and exchange information as a result of the fast development of communication through the internet in the past two decades. Through different social media platforms, online discussion forums, blogs, message boards, product evaluations, and comments on products, people express their opinions.

All of this data may be analysed to better understand human behaviour and to inspire people to work toward certain goals. For instance, we can determine if a product is successful or a failure by examining customer evaluations and comments, and we can then develop the best marketing strategy to support it. Another example would be to offer sentiment analysis of people's blog and message board posts, which might be used to sway people's decisions. Text data analysis includes all such analyses of text-based data.

Text analysis is the process of searching through unstructured and semi-structured text data for meaningful insights, trends, and patterns. The text data analysis algorithm would perform better than any person in the cases above in terms of effectiveness, information quality, and speed. Less resources are required, and the outcomes are less skewed. This makes providing a qualitative and quantitative insight of their goods, consumers, and market very desired to numerous organisations across various sectors.

Data visualisation technologies are used with information gleaned from text data analysis to create actionable insights that support decision-making. Text analysis uses a variety of techniques, including sentiment analysis, document categorization, supervised classification, clustering, document feature identification, natural language processing, and quantitative text summarization. One of them is text categorization, in which the algorithm's goal is to group the text into one or more pre-established categories. Examples of text categorization include:

• Using social media to determine the mood of the audience.

• Email spam and non-spam detection.

• Automatic tags are applied to customer enquiries.

• News items are grouped into predetermined themes.

• Find out if the article is authentic or not.

Although text classification's power looks exciting, there are several possible issues. Text is one of the most prevalent forms of unstructured data, which makes up around 80% of all data. Due to the unstructured nature of text data and the difficulty and time required to manually analyse it, many businesses do not fully utilise the information. Text classification and machine learning are used in this situation. Text classifiers may be used by businesses to swiftly and efficiently organise all relevant material, including emails, legal documents, social media posts, chatbot conversations, and survey responses. Due to this technology, businesses may automate business procedures, save time on text data analysis, and make data-driven business decisions.

Traditionally, text classification can mainly be broken down into four major parts:

1. Feature Extraction
2. Dimensionality reduction
3. Classification techniques
4. Evaluation

Various developments have been done in these four areas of text classification and they also have provided decent results. However, these methods are not 100% accurate and pose certain qualitative problem in the pipeline. Firstly, each of these algorithms analyse text syntactically. i.e., these algorithms analyse each text individually and doesn't consider any relationship among the texts that are in the same class. This leads to a loss of important information about the relationship among the documents. Secondly, these algorithms seldom look for the context in the data or in other words, "meaning between the lines" of text. By understanding the context of the information in each class, we can provide additional attributes for classification that will strengthen the overall outcome of the process.

To cater the two challenges mentioned in the previous section, we propose a novel solution for understanding the context behind the texts in the classification process. We introduce a clustering step after dimensionality reduction step in traditional text classification process. This step acts as a text pre-processing step and identifies the connections of the words that appear frequently together in the text. By grouping words in clusters, we can get a contextual metric and hence boost text classification.

Thus, the objective of this report is to understand the impact of introducing clustering of text as a pre-processing step in text data classification. The problem that we are addressing is as follows:

- ➢ Given a binary class text dataset, provide a classification algorithm that can outperform previously implemented methods for accurately classifying documents.
- ➢ Implement a text clustering method as a pre-processing step of text classification to improve overall accuracy and precision of the process.

II Background Information of Text Classification

2.1 Literature Review

Himank Gupta et. al. gave a structure in view of various AI approach that arrangements with different issues including precision lack, delay (BotMaker), and high handling time to deal with a huge number of tweets in 1 sec. **Shivam B. Parikh** means to introduce a knowledge of portrayal of the report in the advanced diaspora joined with the differential substance kinds of the report and its effect on perusers. In this way, we plunge into existing phony news recognition moves that are vigorously founded on text-based examination, and furthermore portray famous phony news datasets. We finish up the paper by distinguishing 4 key open exploration challenges that can direct future examination. It is a hypothetical Approach that gives Illustrations of phony news identification by investigating the mental elements The information is unstructured; mining the information prompts finding significant opinions about different substances through proper order procedures. In this paper, tweets' viewpoints are broken down through AI calculations, for example, guileless Bayes and backing vector machines utilizing R programming; results are figured out and analyzed. The SVM model shows higher accuracy, and innocent Bayes gives higher exactness to opinion investigation on the Bangalore traffic data. **Sikha Bagui and Debarghya Nandi at. al**. (2021) Representation of message is a critical errand in Natural Language Processing (NLP) and lately, Deep Learning (DL) and Machine Learning (ML) have been broadly utilized in different NLP undertakings like subject characterization, feeling examination and language translation.[2] A correlation of different boundaries and hyper parameters was performed for DL. The consequences of different ML models, Naïve Bayes, SVM, Decision Tree, as well as DL models, Convolution Neural Networks (CNN) and Long Short Term Memory (LSTM), was introduced. **Himank Gupta et. al**. gave a system in view of various AI approach that arrangements with different issues including precision lack, delay (BotMaker), and high handling time to deal with a large number of tweets in 1 sec.

The issue of recognizing not-certified wellsprings of data through happy-based examination is viewed as reasonable in the area of spam recognition [7], spam location uses measurable AI procedures to group text (for example tweets [8] or messages) as spam or authentic. These procedures include pre-handling of the text, highlight extraction (for example pack of words), and element determination in light of which highlights lead to the best exhibition on a test dataset. When these elements are acquired, they can be grouped utilizing Naive Bayes, Support Vector Machines, TF-IDF, or K-closest neighbors classifiers. These classifiers are normal for managed AI, implying that they require a piece of marked information to get familiar with the capability III. Proposed System Methodology

3.1 Performance Metric

Accuracy

Model Accuracy is the most common metric used for performance evaluation of classification and regression models. Accuracy is defined as the measure of correctness of the model predictions.

Mathematically:

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Number\ of\ all\ predictions}$$

Although accuracy provide enough information about the performance of the model, it is more of a summarised score rather than being detailed. For example, if a model performs really well in classifying only one type of class and performs poorly in classifying other classes, then it is not a good classification model. Therefore, a better metric needs to be considered.

Confusion matrix

Consider the matrix below:

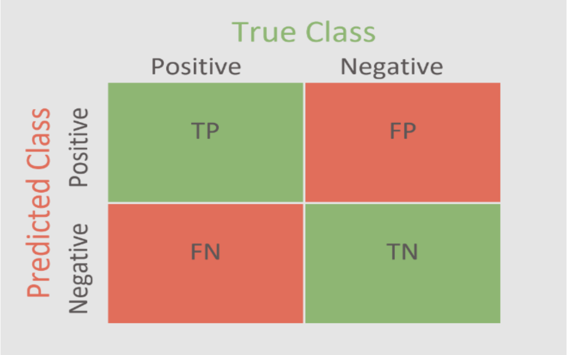

Figure 1 Sample Confusion Matrix

This is called a confusion matrix. It may not be an evaluation metric on its own, but it can provide deeper understanding of performance of the model. There are 4 main terms in a confusion matrix which are defined as follows:

- True Positive (TP): Number of data points predicted true that are actually true.
- True Negative (TN): Number of data points predicted False that are actually False.
- False Positive (FP): Number of data points that are predicted True that are actually False.
- False Negative (FN): Number of data points that are predicted False that are actually True.

Based on these terms we define following metrics

Precision

Precision is the measure of how good the model can predict a particular class (Generally the first Class). It is defined as:

$$Precision = \frac{TP}{TP + FP}$$

Recall

Recall measures that out of all the predictions that are marked positive, how many of them are actually positive. In other words, it measures the completeness of model predictions. Mathematically:

$$Recall = \frac{TP}{TP + FN}$$

A good model is the one with high precision and high recall. However, precision and recall cannot be increased simultaneously and thus, there needs to be trade-off. A new metric is defined as follows.

F1-Score

F1-score is the weighted average of precision and recall and is defined as

$$\boldsymbol{F1\_score = 2\frac{Precision * Recall}{Precision + Recall}}$$

F1-score is a better metric to evaluate model performance when the data is skewed towards one class or the other.

3.2 Proposed Methodology

3.2.1 Challenges identified in current approaches

So far, we have discussed the approaches that are used for text classification use cases. Most of them work well with binary classification and provide decent results. However, a potential drawback is identified which is not handled by any of the algorithms mentioned in previous sections. Most of the pre-processing steps ignore the semantic information from the text data due to which the context of the information is lost. By analysing the context of the document and associating it with a class, it would be easier to classify them. One way to understand the context behind the text is to identify the connections of the words that appear frequently together in the mail. By grouping words in clusters, we can get a contextual metric and hence boost text classification.

3.2.2 Proposed Algorithm

We propose a text clustering based pre-processing step that will analyse the words that are used in the document and will analyse which data points are near to each other. By understanding the position of data points in the vector space, we can understand which documents are closer to each other in terms of words used in them. The scattered data points will then be assigned into two groups or clusters which is closest to them.

This way a new attribute will be assigned to the original features. It is assumed that the documents that have similar vector space (in other words, have similar words appearing in them) will have similar classes. Therefore, by understanding the context from all the documents present in the data, we can build a better classification model.

3.3 Algorithm Flow chart

This figure provides a pictorial representation of the proposed algorithm in the project. First, the text data is loaded in the environment. Next, a few basic pre-processing steps are performed on the data. These include separating text attribute from the labels, removing punctuations, decapitalizing, dropping nulls, removing stop words and special characters,etc.

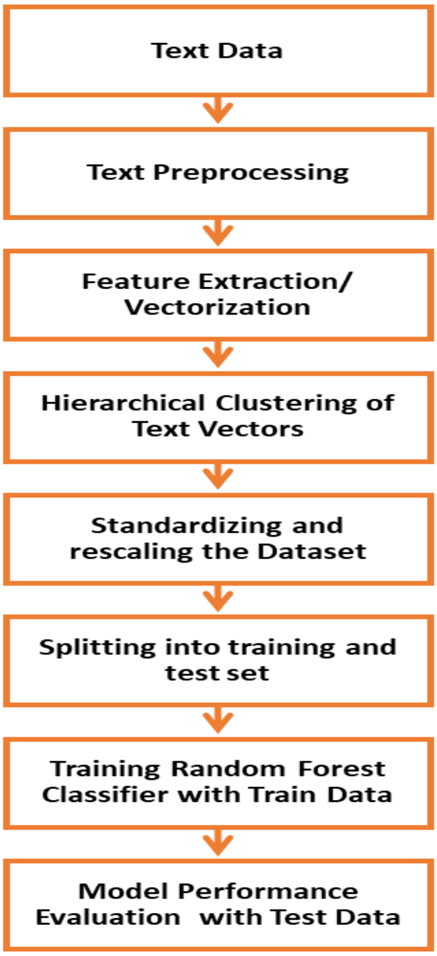
Text Data
Text Preprocessing
Feature Extraction/ Vectorization
Hierarchical Clustering of Text Vectors
Standardizing and rescaling the Dataset
Splitting into training and test set
Training Random Forest Classifier with Train Data
Model Performance Evaluation with Test Data

Figure 1: Test Classification New Approach

Then, we perform TF-IDF vectorization, where each row of text data is converted into the vector format. Here, each vector represents the frequency of a word that appears in the text. After vectorization, the data is subjected to K-means clustering where the clusters are formed based on the vectors obtained in the previous step. The cluster label for each vector is used as a new attribute and is added as a new feature in the dataset. The final modified dataset is split into training and test set. Training set is used to train the ensemble classification model (Random Forest). Validation set (Taken from the training set) is used to fine tune the model. Once the model is trained successfully, the test data is used to evaluate the performance of the model. This performance is compared with the base algorithm (without clustering) and the observations are done.

Algorithm Steps

1. Tokenising the text using stemming.
2. Transforming the text corpus into vectors using TF-IDF.
3. Using Hierarchical Clustering techniques to cluster the texts into two sets.

4. Using the new clusters as a feature along with the text vectors.
5. Splitting the modified dataset into training set and the test set.
6. Training the classification algorithm with the training set and evaluating its performance on the test set.
7. Observing and reporting the results using k-fold Cross validation techniques.

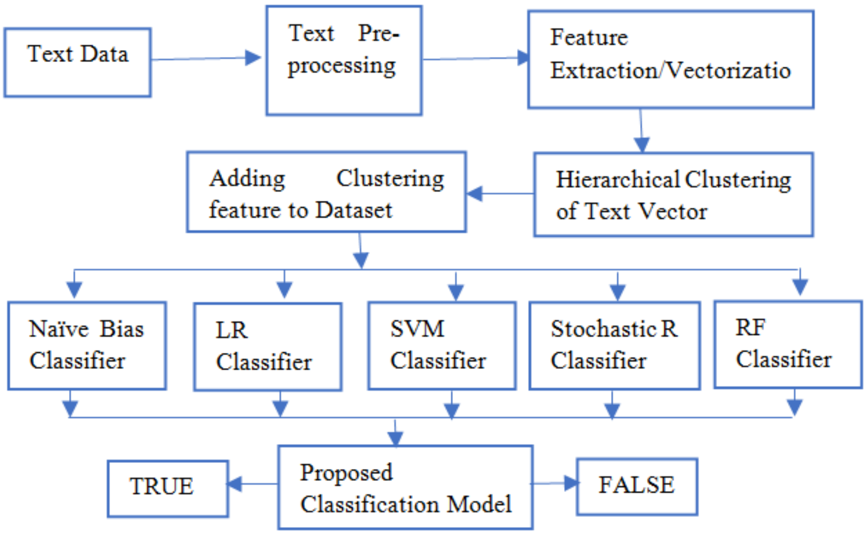
Text Data
Text Pre-processing
Feature Extraction/Vectorizatio
Hierarchical Clustering of Text Vector
Adding Clustering feature to Dataset
Naïve Bias Classifier
LR Classifier
SVM Classifier
Stochastic R Classifier
RF Classifier
TRUE
Proposed Classification Model
FALSE

Figure 2: Proposed Algorithms Flow

IV Proposed Algorithms Results

This work is divided into two main parts:

1. Part 1: Text pre-processing.
2. - Getting the data into the platform.
3. - Extracting the relavent text columns and labels.
4. - Cleaning and normalising the text data into word vectors.

- Clustering the data and creating a new feature.

5. Part 2: Classification using Random Forest algorithm.
6. - Creating 4 different Datasets for comparisons of base data and added cluster features.
7. - Training the Random Forest Classification algorithm on all 4 datasets.

- Testing the classification model with test data and saving the results.

V. Conclusion

Text analytics and NLP can be used to work with the very important problem of text classification. We have seen the big impact they can have on people's opinions, and the way the world thinks or sees a topic. We've built a machine learning model using sample data for detecting fake material, but the process is very similar to detect fake news or anything like that.

References

[1] Annie Syrien and M. Hanumanthappa at.al "Evaluation of Supervised Classification Techniques on Twitter Data using R" International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075 (Online), Volume-10 Issue-8, June 2021

[2] SikhaBagui and Debarghya Nandi at. al. "Machine Learning and Deep Learning for Phishing Email Classification using One-Hot Encoding" Journal of Computer Science 2021, 17 (7): 610.623 , DOI: 10.3844/jcssp.2021.610.623

[3] P. Arumugam and V. Kadhirveni at. al. “Prediction, Cross Validation and Classification in the Presence COVID-19 of Indian States and Union Territories using Machine Learning Algorithms” International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-10 Issue-1, May 2021

[4] Vijay and Dr. PushpneelVerma “Linear Discriminant Analysis for Hate Speech Text Classification” International Journal of Engineering Development and Research ISSN: 2321-9939 , Volume 9, Issue 2 ©IJEDR 2021 Year 2021

[5] Hao Zhang and Yanchun Liang at. al “Deep Feature-Based Text Clustering and Its Explanation” Journal of Latex Class Files, vol. 14, No. 8, Jul 2020, DOI 10.1109/TKDE.2020. 3028943, IEEE

[6] Anastasiu, D. C., Tagarelli, A., Karypis, G. Document Clustering: The Next Frontier. In Aggarwal, C. C. & Reddy, C. K. (Eds.), Data Clustering, Algorithms and Applications (pp. 305–338). Minneapolis: 2014 Chapman & Hall.

[7] Bouguettaya, A., Yu, Q., Liu, X., Zhou, X., & Song, A., Efficient agglomerative hierarchical clustering. Expert Systems with Applications, 42(5), 2015 2785–2797.

[8] Wang, G., Zhang, X., Tang, S., Zheng, H., & Zhao, B., Unsupervised clickstream clustering for user behavior analysis. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (p. 225–236).

[9] Jernite, Y., Bowman, S. R., ,& Sontag, D., Discourse-based objectives for fast unsupervised sentence representation learning. CoRR, 2(2), 2017 758-786.

[10] Agrawal, A., & Gupta, U.; Extraction based approach for text summarization using k-means clustering. In Proceedings of the international conference on information and knowledge management 2014 (Vol. 4, p. 9–12).

[11] Mikolo, T., Chen, K., Corrado, G., & Dean, J.,Efficient estimation of word representations in vector space. In In proceedings of workshop at iclr 2013 (Vol. 1, p. 89–152).

[12] Anita Kumari Singh, MogallaShashi. Vectorization of Text Documents for Identifying Unifiable News Articles , IJACSA 2019 (Vol. 10, No. 7, P 305-310)

[13] Conroy, N., Rubin, V. and Chen, Y., Automatic deception detection: Methods for finding fake news‖ at Proceedings of the Association for Information Science and Technology, 52(1), 2015 pp.1-4.

[14] AayushRanjan,. Fake News Detection Using Machine Learning‖, Department Of Computer Science & Engineering Delhi Technological University 2018

[15] S. B. Parikh and P. K. Atrey "Media-Rich Fake News Detection: A Survey," IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), Miami, FL, 2018, pp. 436-441

[16] V. L. Rubin, Y. Chen, and N. J. Conroy, “Deception detection for news: three types of fakes,” Proceedings of the Association for Information Science and Technology, vol. 52, no. 1, 2015 pp. 1–4.

[17] S. B. Neel Rakholia “is it true? - deep learning for stance detection in news,” tech. rep., Stanford Uiversity, 2017.

[18] O. Papadopoulou, M. Zampoglou, “A two-level classification approach for detecting clickbait posts using text-based features,” preprint arXiv:1710.08528 2017.

[19] Reis, J. C., Correia, A., Murai, F., Veloso, A., Benevenuto, F., & Cambria, E. (2019). Supervised Learning for Fake News Detection. IEEE Intelligent Systems, 34(2), 76-81.

[20] Pal, S., Kumar, T. S., & Pal, S. (2019). Applying Machine Learning to Detect Fake News. Indian Journal of Computer Science, 4(1), 7-12.