Fine-Tuning Pipeline: A Strategic Approach to Multiclass Text Classification

Veerababu Reddy*[1] and N Veeranjaneyulu[2]

[1] Department of CSE, Vignan's Foundation for Science Technology & Research. Vadlamudi 522213, INDIA
*veerababureddy@gmail.com
[2] Department of IT, Vignan's Foundation for Science Technology & Research. Vadlamudi 522213, INDIA
veeru2006n@gmail.com

Abstract. In present days, many industries are incorporating text classification in their ordinary tasks for consistency, scalability, and timeliness it brings in. However, while working with real data there are several obstacles to overcome during conducting the classification modeling. Natural Language Processing (NLP) is the key component in the extraction and analysis of textual data. Multi-Class Text Classification (MCTC) plays a vital role in categorizing textual data into more than two predetermined classes to avoid ambiguity. Most of the issues faced by previous MCTC models are imbalanced datasets, selecting appropriate algorithms, ensuring model generation, and working with high-dimensional data. A model is developed to overcome these challenges by applying Prompt Engineering, NLP pipeline algorithm, and fine-tuning methods along with GPT-3.0 for the existing MCTC model. This model achieved better results with performance metrics of 0.85 F1 score which shows the best accuracy when compared with all other models, including BERT, Bart, and GPT.

Keywords: Natural Language Processing, Multi class Text Classification, Prompt Engineering, Pipeline, Fine Tuning, F1 score.

1 Introduction

In natural language processing (NLP), multiclass text classification is an essential job that has applications ranging from subject categorization to sentiment analysis. This work introduces a unique method that integrates using our model, an advanced fine-tuning methodology, to improve multiclass text classification performance.

Multiclass text classification is essential in question-answer generation as it categorizes input text into various classes or topics, providing a structured framework for generating relevant questions. By accurately classifying text, the question generation process can be guided to focus on specific themes or subjects present in the input. Multiclass text classification serves a crucial purpose in question-answer generation, such as topic identification, quality assurance, customization, and personalization.

K. R. Madhavi et al. (eds.), *Proceedings of the International Conference on Computational Innovations and Emerging Trends (ICCIET 2024)*, Advances in Computer Science Research 112,
https://doi.org/10.2991/978-94-6463-471-6_90

Additionally, multiclass text classification ensures diversity in question generation by covering a wide range of topics represented in the input text. This diversity leads to a more comprehensive set of questions, catering to different aspects of the content. Furthermore, multiclass text classification aims at personalization, allowing question generation systems to customize questions based on user preferences or specific requirements. Overall, multiclass text classification serves as a foundational step in question-answer generation, facilitating the production of relevant Question and Answers.

Most research has been done on multi-class text classification using traditional machine learning algorithms such as KNN, SVM, etc. Based on these studies, class classification becomes difficult because the training and testing model takes too much time. The traditional methods produce very inaccurate results, and the power consumption is too high. Performing multi-class text classification using traditional methods also requires high-configuration systems.

Based on the improvement in technologies, the classification tasks are carried out with deep learning algorithms such as CNN, RNN, etc., where we find increased accuracy but no change in time complexity. The process of training and testing these models also became a very challenging task.

Past research on multiclass text classification was done using multiclass convolutional neural networks. The research was different compared to others where CNNs with a single input, single processing, and a single convolution layer were implemented and also introduced an extra layer called the multi-class layer for classification.

In our research, the model was developed using LLMS like GPT. Fine-tuning was carried out in the process to get the desired outcomes. Using the fine-tuning process in the NLP pipeline, our model has achieved superior accuracy compared to other models. Based on these results, our model has become domain-independent.

2 Literature Survey

Ankita Dhar et al. [1] TF-IDF algorithms are used for the classification of text documents into multiple classes the algorithms are based on the distance measurement between the classes and object the proposed model was capable of providing outputs only in a specific language

Xian She et al. [2] Proposed a softmax aggression algorithm for text classification the performance was improved to a greater extent such that our model produces the desired outcomes. The softmax regression algorithm has improved the classification accuracy rate hour for the nonlinear text the software regression algorithm helps in a deeper understanding of the data and classifying the classes into their most relevant classes it was created in a way such that it can handle only one type of data at a time.

W.Wu. [3] Proposed a new method by combining feedforward neural networks with another method called the Binary approach. This method was most useful with classification classes restricted to a lesser number the mortal performs well when the classification problems are up to five classes the algorithm is based on n-class classification where the given data is divided into N classes.

S. Yu and J.Su. [4] Proposed a new methodology by using the models BERT and BERT4TC, and the model is a BERT-based model. First, the model was undergone with vigorous training of the data sets available and the model was tested. The model performed well according to the data set used in the training and the difficulty in it. It worked well for the limited number of domain data sets.

Chen and H. Chen. [5] A novel class center vector model was introduced for text classification. class center vector method and TF-IDF are used for training and testing the model, the accuracy of the model was dependent on the number of layers used in the model. The model performed well up to six clustered layers the clustered layers are used for extracting the relation between the classes.

K. Xia. [6] Proposed a methodology that combines various algorithms for the classification of text. The model includes encoders, decoders multi-label text classification, and hierarchical text classification algorithms. The proposed approach outperforms the baseline approach in handling the classification of police case texts. This model was extended to only specific domains.

Hui Gao. [7] Combined convolution neural network (CNN) blatant Dirichlet allocation (LDA) for appropriate classification of reviews the model was used for non-temporal data and CNN was used for feature extraction or class extraction from data.

K. A. Qureshi. [8] Implemented a method for social media-based multi-class hate speech classification for text the support vector machine (SVM) logistic regression and multilayer perceptron are used in this model. The result produced by our model has shown significant accuracy it plays a major role in the classification of hate speech.

Khan et al. [9] Proposed a model MCNN-LSTM attaching CNN and LSTM to classify multi-class text classification in imbalanced news data sets in this method the layers in the CNN are changed and limited to one layer on each type. LSTM is used for extracting long-term features from text dramatic link algorithm is used for balancing the imbalance data sets it can learn phrase-level features and characteristics to learn long-term dependencies

S. Joshi et al. [10] Multi-class text classification using machine learning models for online drug reviews used a machine learning algorithm. It is found to be more efficient in predicting the medical conditions regarding the accuracy and it was limited to the medical domain.

3 Existing System

The existing was implemented with machine learning models that don't produce accurate results. The machine learning thresholding, K-means, and support vector machine (SVM) algorithms were used. Disadvantages of Existing Models:

Handling Imbalanced Data: There is a need for research focusing on techniques to effectively handle imbalanced data to ensure fair and accurate classification across all classes.

Transfer Learning: While transfer learning has shown promise in improving performance in various NLP tasks, its application to multiclass text classification has not been extensively explored.

Domain Adaption: Text classification models often struggle when applied to domains different from the one they were trained on. Research gaps exist in exploring domain adaptation techniques tailored specifically for multiclass text classification, especially in scenarios where labeled data in the target domain is limited.

Incorporating External Knowledge: There is a gap in research focusing on methods to effectively incorporate external knowledge into multiclass text classification models to improve performance and generalization.

4 Methodology

The Methodology implemented in this project involves a systematic integration of deep learning techniques and NLP techniques the code snippet produced in the notebook serves as the practical implementation of the outlined methodology.

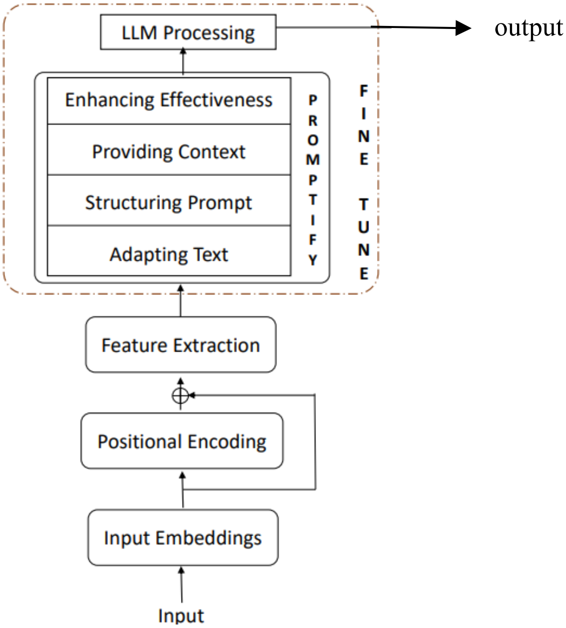

Fig. 1. The architectural diagram

Data collection: The first step in the model was beta collection and it was done by gathering relevant text data from various sources such as weblinks databases etc. The collected data will be either balanced data or imbalanced data.

Data Preprocessing: Rigorous preprocessing steps are implemented to standardize the input data and enhance the model's generalization. The preprocessing ensures that the neural network receives uniform input, thereby improving its ability to discern intricate patterns. Data Balancing: If any imbalanced data sets are present then these data sets are balanced using the Tomek link algorithm the balanced data sets give more accurate results when compared to the imbalanced data sets. Feature Extraction: Feature extraction can be done by implementing some of the encoding and decoding techniques in feature extraction the relation between the classes is extracted. Encoder: It is used for blasting textual data into numerical data by catching the semantic meaning and properties the representations can be fed into a classification algorithm to predict the class label. Decoder: the decoder helps in generating tokens and upon each generation represents tokens relying on previously generated tokens and input representations until the entire output sequence is produced. Implementation: Implementation is one of the major steps in our process where the pre-trained LLMS are used for producing the output. There are different stages involved in this stage the models are utilized using the Transformers architecture for their work. Transformers are updated deep-learning models for sequential data. Fine tuning: Fine tuning is the major step in our process to get desired outputs we can perform fine tuning on our model where it can perform well fine tuning helps in getting more accurate results and also helps in extracting classes more accurately.

In summary, the model is widely used in classification tasks in NLP problems from classifying movie reviews to classification of texts. The main usage of this model was in Automatic Question-Answer generation. The QAG system reduces the time for many professors to generate questions and answers.

Model Evaluation: The model is evaluated on different text modules to assess its performance in terms of accuracy and F1-Score.

Prediction on Custom Input: The trained model is used to predict the class of a custom input, providing insights into the model's real-world applicability.

5 Discussion

The model has undergone vigorous testing with different domain data testing texts. Based on the results it is made that our model is independent of the domain and produces results for every domain textual data. F1 score is used to calculate accuracy and the above table is taken as a sample. Our model solves multi-class classification tasks with each and less time complexity. Unlike the existing models takes too much time for training and testing and the results produced aren’t accurate.

Table 1. F1-Score evaluation on different textual data

Texts	Fine - Tuning
T1	0.83
T2	0.86
T3	0.81
T4	0.89

The model has undergone vigorous testing with different domain data testing texts. Based on the results it is made that our model is independent of the domain and produces results for every domain textual data. F1 score is used to calculate accuracy and the above table is taken as a sample. Our model solves multi-class classification tasks with each and less time complexity. Unlike the existing models takes too much time for training and testing and the results produced aren't accurate.

Fine-tuning was carried out through a process for accurately predicted output and the Fine-tuning process was applied immediately if the output was not satisfied until we got the desired output.

Table 2. Comparison of our model with existing models

Texts	BERT	BART	Gemini	GPT	Fine-Tuning
T1	0.34	0.45	0.67	0.74	0.83
T2	0.17	0.44	NA	0.71	0.86
T3	0.25	0.57	NA	0.77	0.81
T4	0.39	0.38	0.69	0.79	0.89

The model accuracy was compared to the previously existing models such as BERT, BART, GPT, etc. The initial models have produced very less accurate results and some of the developed AIs are unable to produce the desired outcomes so using the existing models like LLMS we introduced the fine-tuning methodology to improve the performance of the previously existing model. Using the fine-tuning methodology our model outperforms well when compared to the existing model.

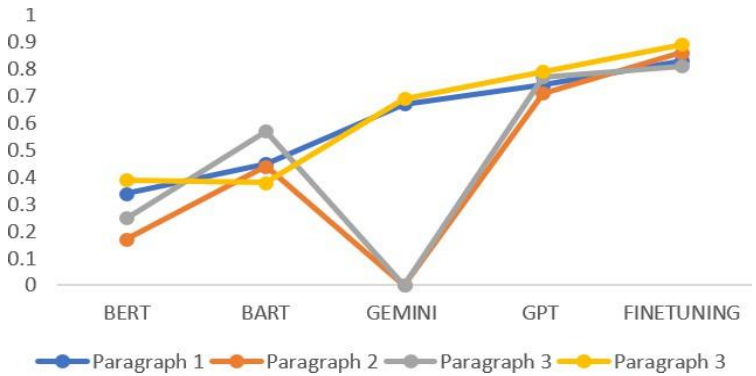

Fig. 2. Graphical representation of evaluated values of different models.

The above graph is the comparison of our model with different models on different texts. From the graph, it is observed that our models show superior accuracy than any other model. The graph also describes how each model responds to the same text given and shows the F1-Score for each model. Some of the models were not capable of producing results and the outcomes from the GEMINI model were not satisfying.

The GPT model has shown some closeness to our model in terms of output, but the results were not much more accurate When compared to our model. The initial models are very less accurate in terms of output and the time taken for their training and testing.

Fine-tuning was carried out using more efficient algorithms and models like encoders, decoders, transformers, etc. These Models are responsible for weight updating and extracting the most relevant features from the text we are using. So fine-tuning helps in getting more accurate results when compared to the other models.

6 Conclusion

The proposed model evolved through the fine-tuning of GPTs and LLMs. To enhance its capabilities, we incorporated fine- tuning technology, which facilitated the execution of various intricate tasks. Notably, the library's pipe fit module enabled the model to effectively extract entities and relationships, further augmenting its proficiency in handling diverse text processing requirements.

While previous attempts have tackled similar tasks, they often struggled with low accuracy rates. However, our proposed model has achieved a notable 80% accuracy, representing a significant performance improvement. This enhanced accuracy underscores the effectiveness of our approach, distinguishing it from previous endeavors and highlighting its potential for robust and reliable text classification tasks.

As our model is built upon the foundation of GPTs and LLMs, future advancements in this technology hold promise for further improvements. With the introduction of new iterations, the model's accuracy and capacity to handle diverse datasets may see significant enhancements.

7 Future Work

Investigate the application of advanced neural network architectures, such as transformer-based models (e.g., BERT, GPT, XLNet), for multiclass text classification tasks. These models have shown remarkable performance in various natural language processing tasks and could potentially improve classification accuracy and generalization capabilities. Explore methods to incorporate external knowledge sources, such as knowledge graphs or domain- specific ontologies, into the text classification model. This could help the model better understand the context and semantic relationships within the text, potentially improving classification performance, especially in specialized domains. Explore transfer learning and domain adaptation techniques to leverage knowledge from pre-trained models or related domains. This could improve the model's performance on target domains with limited labeled data and facilitate faster adaptation to new tasks or domains. Develop new evaluation metrics and benchmarks tailored specificallyfor multiclass text classification tasks. Existing metrics may not capture all aspects of performance, and new benchmarks could facilitate more robust and comprehensive evaluation of different models and techniques.

References

1. Ankita Dhar, NiladriSekhar Dash, and Kaushik Roy, “Classification of Text Documents Through Distance Measurement: An Experiment with Multi-Domain Bangla Text Documents”, VOLUME 11, 2017.
2. Xiangyang She, and Yinglong Zhu, “Text Classification Research Based on Improved Soft-Max Regression Algorithm”, VOLUME 8,2018.
3. Sibo Yang, Chao Zhang, and Wei Wu, “Binary Output Layer of Feedforward Neural Networks for Solving Multi-Class Classification Problems”, IEEE Access Multidisciplinary, Rapid Review, Open Access Journal, Vol.7, 2019, 5085-5094.
4. Shanshan Yu and Jindian Su, “Improving BERT-Based Text Classification with Auxiliary Sentence and Domain Knowledge”, IEEE Access Multidisciplinary, Rapid Review, Open Access Journal, Vol.7, 2019, 176600-176612.
5. Xinhua Zhu, Qingting Xu, Yishan Chen, Hongchao Chen, and Tianjun Wu1, “A Novel Class-Center Vector Model for Text Classification Using Dependencies and a Semantic Dictionary”, IEEE Access Multidisciplinary, Rapid Review, Open Access Journal,2019, 24990-25000.
6. Kumar, DNS Ravi, N. Praveen, Hari Hara P. Kumar, Ganganagunta Srinivas, and M. V. Raju. "Acoustic Feedback Noise Cancellation in Hearing Aids Using Adaptive Filter." International Journal of Integrated Engineering 14, no. 7 (2022): 45-55.
7. Naila Aslam, Waheed Yousuf Ramay, Kewen Xia, and Nadeem Sarwar,” Convolutional Neural Network Based Classification of App Reviews”, IEEE Access Multidisciplinary, Rapid Review, Open Access Journal,2020, 185619- 185628.

8. Khubaib Ahmed Qureshi and Muhammad Sabih, "Un-Compromised Credibility: Social Media Based Multi-Class Hate Speech Classification for Text", IEEE Access Multidisciplinary, Rapid Review, Open Access Journal,2021, 109465- 109477.
9. Xiangyang She and Yinglong Zhu, "Text Classification Research Based on Improved Soft-Max Regression Algorithm", VOLUME 8,2018.
10. Khan Md Hasib, Sami Azam And Asif Karim, "MCNN-LSTM: Combining CNN and LSTM to Classify Multi-Class Text in Imbalanced News Data", VOLUME 11,2023.
11. Shreehar Joshi and Eman Abdelfattah, "Multi-class Text Classification Using Machine Learning Models for Online Drug Reviews", VOLUME 3,2021.