International Scientific Journal of Engineering and Management (ISJEM)
Volume: 04 Issue: 06 | June – 2025
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata
ISSN: 2583-6129
DOI: 10.55041/ISJEM04058

Enhancing Text Classification Using Advanced Natural Language Processing Techniques

Archana H
Data Science and Computer Applications
Manipal Institute of Technology
Manipal, India
archana.suk@manipal.edu

Brijesh Jaya Poojary
Data Science and Computer Applications
Manipal Institute of Technology
Manipal, India
brijesh.mitmpl2023@learner.manipal.edu

Chayanka
Data Science and Computer Applications
Manipal Institute of Technology
Manipal, India
chayanka.mitmpl2023@learner.manipal.edu

***Abstract*—This study evaluates the performance of various neural network models and a pre-trained transformer model in the task of quote classification. The models analyzed include Fully Connected Neural Networks (FCNN), Long Short-Term Memory (LSTM), Gated Recurrent Units (GRU), Bi-directional LSTM (BiLSTM), and DistilBERT. The goal is to identify the most effective model for the given dataset based on key performance metrics such as accuracy, precision, and recall. DistilBERT, a lightweight transformer-based model, is also assessed for its efficiency and accuracy compared to traditional neural network approaches.**

***Index Terms*—Natural Language Processing (NLP), Data Preprocessing, Machine Learning, Quote Classification, Multi-label Classification, Neural Networks (NN), Transformer Models, DistilBERT.**

I. Introduction

A key component of artificial intelligence (AI) is NLP, which has enabled machines to effectively understand and process human language. With profound impact across various fields from sentiment analysis and text classification to machine translation, NLP has revolutionized how textual data is handled in domains [1] such as healthcare,social media and customer service.

Recent advancements have enabled machines [2] to understand and produce human language in previously unthinkable ways. A key application, as noted by Wankhade et al. [3], is text classification, which sorts text into categories like sentiment, intent, or topic. NLP's importance has grown, especially for tasks like text classification, crucial for applications such as [4], [5] sentiment analysis, spam detection, and content categorization.

Traditional methods like Term Frequency-inverse Document Frequency (TF-IDF) [6] and Bag-of-Words(BoW) [7] helps in transforming text to numericals aiding the models to learn and understand better. However, these methods perform well for basic text processing but struggle with capturing semantic [8] relationships, context, and nuances in language

To overcome these challenges, **NN** more specifically, **Pre-Trained Transformer Models** have emerged as a powerful tool [9] for text classification. Unlike the traditional models these require large datasets [10] and deep learning architecture to extract complex learning patterns, thus enabling in better understanding to text.

Pre-trained transformer models are neural networks [11] that learn from a lot of text using deep learning.

Among these models, BERT has set new benchmarks in NLP by excelling at contextual understanding. However, its computational demands pose practical challenges, particularly for real-time applications [12]. This has led to the development of more efficient alternatives, such as **DistilBERT**, which retains strong performance while reducing

International Scientific Journal of Engineering and Management (ISJEM)
Volume: 04 Issue: 06 | June – 2025
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata
ISSN: 2583-6129
DOI: 10.55041/ISJEM04058

resource requirements.

Among the various NLP tasks, **Quote Classification** is an intriguing [13] and challenging one. Categorizing quotes to meaningful labels such as sentiment, topic or context requires advanced NLP models which are capable of understanding both individual words and their broader contextual meaning.

This task becomes difficult because human languages are complex. For example, the phrase *"I'm fine"* can mean different things:

- **Literal:** The person is okay.
- **Implied:** They might not be fine but don't want to talk about it.

This ambiguity underscores the need for [14] advanced models that can analyze sentiment, tone, and deeper contextual cues. Transformer-based models provides a wise solution to overcome these problem, particularly those optimized for efficiency

This paper looks at how pre-trained models can make quote classification more accurate and efficient. The goal is to evaluate models that understand subtle language details while staying efficient.

II. Literature Review

Quote classification has become an important task in NLP, especially for applications such as sentiment analysis, topic modeling, and content recommendation systems.

Over the past few years, there has been significant advancements in field of NLP, particularly in tasks like text classification.A prominent approach tha has been gaining lot of popularity is the use of deep learning models, particularly [15] neural networks. NN have proven to be highly effective for a variety of NLP tasks, such as [16] spam detection, content categorization and sentiment analysis , due to their ability to learn complex patterns from large datasets.

Recent advancements in deep learning models like BERT and RoBERTa, have expanded the boundaries of NLP by utilizing transformers and attention mechanisms. These models capture deep contextual relationships within text, significantly improving performance in tasks such as quote classification.

Additionally, optimized versions like DistilBERT and other lightweight transformer models have emerged, offering comparable performance with reduced computational costs. These improvements make pre-trained transformer models a powerful choice for large-scale applications where high accuracy and contextual understanding are required. [17].

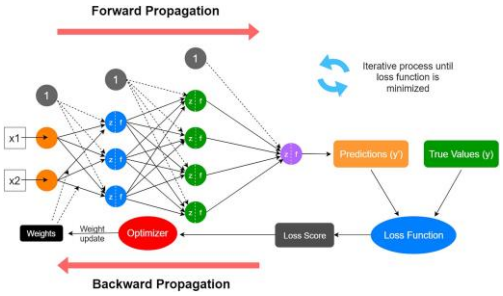

Fig. 1. Neural Network.

Traditional NN's like LSTM, CNN, and GRU handle real-time tasks well, these can recognize patterns without extensive pre-training. On the other hand, newer deep learning models excel at processing large datasets and capturing complex language structures, significantly boosting classification accuracy. The ideal choice depends [18] on the specific needs of the task, with each approach offering unique strengths based on available resources and performance goals.

III. Problem Statement

In recent years NLP has seen remarkable progress, especially in text classification tasks. Traditional machine learning models like Logistic Regression and Naive Bayes laid the foundation for NLP but struggle with language complexities, because they rely on manually created features and miss contextual nuances.

Pre-trained models have transformed text classification by using large datasets to recognize complex language patterns. These models deliver high accuracy, especially with large datasets, making them a powerful choice for many NLP tasks.

NN architectures like LSTM, Convolutional Neural Networks (CNN), GRU continue to remain powerful options. These models effectively capture sequential dependencies and contextual relationships while being more computationally efficient compared to large-scale pre-trained models.

This research aims to enhance text classification by utilizing both pre-trained and neural network-

based approaches. It evaluates their adaptability across different dataset sizes and computational constraints,addressing the limitations of traditional methods, and also contributing to more effective and scalable NLP solutions.

IV. METHODOLOGY

The dataset used in this paper is combination of Quotes Dataset from Kaggle (Kaggle-dataset) and the **Gutenberg Corpus**, which provides a diverse collection of quotes from various domains, including love, life , humour, inspiration etc.

1. Text Classification Process

A. Data Collection

Data collection plays a significant role in any NLP task, as it directly impacts the quality and reliability of model performance.

B. DataSet

For this study, We used the Quotes Dataset from Kaggle, which had significant class imbalance, and supplemented it with the Gutenberg Corpus from NLTK to extract additional quotes, creating a more balanced and diverse dataset for model training.

Both were selected for their rich textual content and their complementary characteristics, enabling analysis across varying contexts and genres of language

The dataset contains $N = 1,39,727$ records with two columns: `quote` (text data) and `category` (target labels).

C. Data Preprocessing

The data from Quotes Dataset and Gutenberg Corpus was combined into one single dataset beforehand and then loaded.

To focus on significant categories, we filtered the dataset to retain only those categories with more than $T = 30,000$ samples:

$$C = \{\text{inspiration}, \text{life}, \text{love}\}.$$

The filtered dataset has the following distribution:

$$\begin{aligned} \text{inspiration} &: 30,379 \text{ samples}, \\ \text{life} &: 30,851 \text{ samples}, \\ \text{love} &: 30,724 \text{ samples}. \end{aligned}$$

1) Text Normalization:

- **Lowercasing**: Converted all text to lowercase to maintain uniformity.

$$w_j \rightarrow \text{lower}(w_j) \quad (1)$$

- **Stopword Removal**: Remove tokens w_j that belong to the set of stopwords S:

$$q_i' = \{w_j \mid w_j \notin S\}. \quad (2)$$

2) Tokenization: Tokenization is the process of dividing text into smaller units/parts called tokens, which could be words, characters or sub-words.

$$q_i = \{w_1, w_2, \ldots, w_k\}. \quad (3)$$

3) Label Encoding: Is applied to the category column to convert them to Numerical format making it easier for the algorithms to process the labels.

D. Feature Representation

Feature representation converts text into a numerical format that is used by machine learning models.

Here we have set `max_features=5000`, therefore the vectorizer limits the representation to the top 5000 most significant words in the dataset, based on their TF-IDF scores.

The processed data was split into training and testing sets with an 80-20 ratio:

$$Q_{\text{train}}, Q_{\text{test}} \subseteq Q_{\text{processed}} \quad (4)$$

2. Model Training

In the model implementation, the input data is reshaped as $(n, 1, d)$:

- n: Number of samples
- 1: Single-time-step data
- d: Number of features

E. Basic Neural Network Models

- **LSTM**
  This model is defined using 128 units of LSTM layer. Then the model is compiled with Adam Optimizer with 0.001 as learning rate which easily adapts to any learning rate dynamically for each parameter.
  Early Stopping has been introduced to stop the training if the model validation loss doesn't show significant improvement after 5 epochs therefore preserving the best weights. The layer

ISSN: 2583-6129
DOI: 10.55041/ISJEM04058

of ReduceLROnPlateau is added so that there will be smooth convergence if validation loss doesn't budge for 3 epochs, which then reduces the learning rate by a factor of 0.5.

$$\eta_{\text{new}} = \eta_{\text{current}} \times 0.5 \quad (5)$$

- **GRU**

The GRU model consists of a layer with 128 units, which processes the input data to find temporal dependencies. Since GRU is a lighter version of LSTM, with fewer parameters, the model can maintain performance while requiring less training time.

The output generated by the GRU layer (h_t) is then passed to 64 neurons in a fully connected layer along with ReLU activation:

$$y_{\text{dense1}} = \text{ReLU}(W_{\text{dense1}} \cdot h_t + b_{\text{dense1}})$$

The layer above is applied with softmax activation, which is best suited for multi-class classification and class probability generation:

$$\text{softmax}(y_i) = \frac{e^{y_i}}{\sum_{j=1}^{C} e^{y_j}} \quad (6)$$

- **FCNN**

FCNN Model is implemented using 128 neurons of dense input layer along with ReLU activation.

L2 Regularization of 0.0001 is added to combat large weights

$$L2 = \frac{1}{2}\lambda \sum W^2 \quad (7)$$

Layer of Batch Normalization is made to normalize the activations to stabilize the training process.

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}} \quad (8)$$

Model is randomly made to drop 30% of neurons to avoid overfitting during training.

An additional of 2 dense layers are introduced: the second layer with 64 neurons, batch normalization and 30% dropout, the third layer with 32 neurons , batch normalization and 20% dropout.

- **BiLSTM**

The BiLSTM model differs from previously trained models due to its unique approach: BiLSTMs can access both past and future context within a sequence.

Following the BiLSTM layer, a Dense layer with 64 neurons is applied, followed by ReLU activation.

- **Ensemble Model**

Four distinct models (LSTM, GRU, BiLSTM, and FCNN) are trained independently on the same dataset. Training and predictions are generated from each individual model to provide the probability of each class for each input sample.

All the above models are optimized using the Adam optimizer, which dynamically adjusts the learning rate during training, ensuring stable parameter updates.

F. Pre-Trained Transformer Model

- **DistilBERT**

DistilBERT retains most of BERT's accuracy while being 60% faster and requiring fewer resources.

Fine-tuned on the dataset for three epochs ($N = 3$), model demonstrated effective learning, as evidenced by a steady decline in training loss. This efficiency allows it to perform well in various NLP tasks while maintaining lower computational costs compared to larger transformer models.

The paper [19] presents a model based on DistilBERT and SHAP to detect COVID-19 misinformation. The model achieved high accuracy (0.972) and AUC (0.993), outperforming traditional machine learning models, and emphasized the importance of model explainability to boost public trust.

3. Performance Evaluation

- **LSTM**

Multiple configurations were made through which we collected various test accuracies and the best yet was of 63.95%. Overall its a standard architecture and it performs well

ISSN: 2583-6129
DOI: 10.55041/ISJEM04058

by capturing long-term dependencies it also maintains a memory of previous inputs . Loss functions implemented here made the model performace consistent for several epochs. It also implies that LSTMs are effective and they learn patterns very well from the input data.

- **GRU**
  The GRU model came up with competitive accuracy of around 65.47%. Being a simplified version of LSTMs which means number of gates are reduced and combined as a single update gate it performed really well. Despite having a good accuracy score they lack behind of furthermore increasing the predictive accuracy of the model since complexity and parameters are less compared to the base LSTM.
- **FCNN**
  This model gave test accuracy of 63.36% which is one of lower results. These models are know to identify simple relationships in the data and reason behind the low accuracy is its lack of ability to capture complex pattern found in sequential data.
- **BiLSTM**
  This model achieved a total overall test accuracy of 62.51%. These models are main designed to capture in both forward and backward context . Having the advantage, this architecture only showed average performance in terms of coping up with unseen data.Model's performance slightly diverges from validation accuracy.This implies BiLSTM is capable of processing sequential inputs but it does well when larger datasets are present.
- **Ensemble**
  This approach achieved accuracy of 68.48%.This model aggregates the predictions by combining their strengths. The final outcome was calculated by minimizing the bias around the models and reducing variance. Ensemble model came out as a well-balanced model at par with performance across different metrics.
- **DistilBERT**
  The DistilBERT model was fine-tuned for three epochs, demonstrating a steady decrease in training loss, from 0.4211 in the first epoch to 0.2886 in the final epoch. The validation loss initially fluctuated but stabilized at 0.5817. The model achieved strong generalization, with the final evaluation showing an accuracy of 82.47%. These results indicate that the model effectively learned patterns from the dataset while maintaining efficiency and robustness.

V. Results

The performance of different models was evaluated based on test accuracy. Table I summarizes the accuracy achieved by each model.

Model	Test Accuracy (%)
LSTM	63.95
GRU	65.47
FCNN	63.36
BiLSTM	62.51
Ensemble	68.48
DistilBERT	82.47

TABLE I
Test accuracy of different models

Overall, DistilBERT emerged as the most effective model, providing a strong balance between efficiency and accuracy.

Fig. 2. Train/Epoch vs Train/Global Step.

The graph illustrates the relationship between the training epoch and the global step during the model's training. In Figure. 2 x-axis represents the "train/global_step," increasing from 0 to approximately 27,000, while the y-axis represents the "train/epoch," ranging from 0 to 3. A generally positive, nearly linear correlation can be seen between the two metrics, as expected during sequential training.

ISSN: 2583-6129
DOI: 10.55041/ISJEM04058

VI. CONCLUSION

In our observations, DistilBERT outperformed other models in both accuracy and efficiency, making it the most suitable choice for text classification. Unlike BERT and RoBERTa, which require significant computational resources and large datasets, DistilBERT delivers impressive performance while operating faster and with fewer parameters. This balance of speed, memory efficiency, and accuracy makes it particularly valuable for resource-constrained environments. While models like BiLSTM and GRU showed promise, DistilBERT excelled in our experiments.

DistilBERT's success lies in its ability to process text efficiently without compromising contextual understanding. The train/global_step graph illustrates its consistent and efficient training cycles, highlighting reduced training time and lower memory usage. This strong performance makes it the best overall model in our experimental setup. While larger models like BERT and RoBERTa may offer advantages with extensive datasets and high computational power, DistilBERT itself has the potential to achieve even greater accuracy when trained on larger datasets, further enhancing its effectiveness.

REFERENCES

[1] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[2] Ali Husnain, Hafiz Khawar Hussain, Hafiz Muhammad Shahroz, Muhammad Ali, and Yawar Hayat. Advancements in health through artificial intelligence and machine learning: A focus on brain health. *Revista Espanola de Documentacion Cientifica*, 18(01):100–123, 2024.

[3] Mayur Wankhade, Annavarapu Chandra Sekhara Rao, and Chaitanya Kulkarni. A survey on sentiment analysis methods, applications, and challenges. *Artificial Intelligence Review*, 55(7):5731–5780, 2022.

[4] Seid Muhie Yimam, Chris Biemann, Shervin Malmasi, Gustavo H Paetzold, Lucia Specia, Sanja Štajner, Anaïs Tack, and Marcos Zampieri. A report on the complex word identification shared task 2018. *arXiv preprint arXiv:1804.09132*, 2018.

[5] Ali Pourzangbar, Mahdi Jalali, and Maurizio Brocchini. Machine learning application in modelling marine and coastal phenomena: a critical review. *Frontiers in Environmental Engineering*, 2:1235557, 2023.

[6] Shahzad Qaiser and Ramsha Ali. Text mining: use of tf-idf to examine the relevance of words to documents. *International Journal of Computer Applications*, 181(1):25–29, 2018.

[7] Wisam A Qader, Musa M Ameen, and Bilal I Ahmed. An overview of bag of words; importance, implementation, applications, and challenges. In *2019 international engineering conference (IEC)*, pages 200–204. IEEE, 2019.

[8] Kamran Kowsari, Kiana Jafari Meimandi, Mojtaba Heidarysafa, Sanjana Mendu, Laura Barnes, and Donald Brown. Text classification algorithms: A survey. *Information*, 10(4):150, 2019.

[9] Yujia Wu and Jun Wan. A survey of text classification based on pre-trained language model. *Neurocomputing*, page 128921, 2024.

[10] Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Gadre, Hritik Bansal, Etash Guha, Sedrick Keh, Kushal Arora, et al. Datacomp-lm: In search of the next generation of training sets for language models. *arXiv preprint arXiv:2406.11794*, 2024.

[11] Nikolaus Kriegeskorte and Tal Golan. Neural network models and deep learning. *Current Biology*, 29(7):R231–R236, 2019.

[12] Shang Gao, Mohammed Alawad, M Todd Young, John Gounley, Noah Schaefferkoetter, Hong Jun Yoon, Xiao-Cheng Wu, Eric B Durbin, Jennifer Doherty, Antoinette Stroup, et al. Limitations of transformers on clinical text classification. *IEEE journal of biomedical and health informatics*, 25(9):3596–3607, 2021.

[13] Serhii Voloshyn, Victoria Vysotska, Oksana Markiv, Ivan Dyyak, Ihor Budz, and Vadim Schuchmann. Sentiment analysis technology of english newspapers quotes based on neural network as public opinion influences identification tool. In *2022 IEEE 17th International Conference on Computer Sciences and Information Technologies (CSIT)*, pages 83–88. IEEE, 2022.

[14] Qian Li, Hao Peng, Jianxin Li, Congying Xia, Renyu Yang, Lichao Sun, Philip S Yu, and Lifang He. A survey on text classification: From traditional to deep learning. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 13(2):1–41, 2022.

[15] Basemah Alshemali and Jugal Kalita. Improving the reliability of deep neural networks in nlp: A review. *Knowledge-Based Systems*, 191:105210, 2020.

[16] Xi Ouyang, Pan Zhou, Cheng Hua Li, and Lijun Liu. Sentiment analysis using convolutional neural network. In *2015 IEEE international conference on computer and information technology; ubiquitous computing and communications; dependable, autonomic and secure computing; pervasive intelligence and computing*, pages 2359–2364. IEEE, 2015.

[17] Neil C Thompson, Kristjan Greenewald, Keeheon Lee, and Gabriel F Manso. The computational limits of deep learning. *arXiv preprint arXiv:2007.05558*, 10, 2020.

[18] Naigang Wang, Chi-Chun Charlie Liu, Swagath Venkataramani, Sanchari Sen, Chia-Yu Chen, Kaoutar El Maghraoui, Vijayalakshmi Viji Srinivasan, and Leland Chang. Deep compression of pre-trained transformer models. *Advances in Neural Information Processing Systems*, 35:14140–14154, 2022.

[19] Jackie Ayoub, X Jessie Yang, and Feng Zhou. Combat covid-19 infodemic using explainable natural language processing models. *Information processing & management*, 58(4):102569, 2021.