Journal of Computer Science

Research Article

Hybrid Deep Learning Models for Text Classification: Performance Evaluation of TriDistilBERT and BiGRU Architectures

Amira Samy Talaat

Computers and Systems Department, Electronics Research Institute, Cairo, 12622, Egypt

Article history
Received: 12-02-2025
Revised: 09-04-2025
Accepted: 10-05-2025

Abstract: Text can be a valuable source of information, but its unstructured nature makes analysis challenging and time-consuming. Machine Learning (ML) algorithms can efficiently analyze and structure text, enabling organizations to automate processes and uncover insights that support better decision-making. This study focuses on applying ML to a classification problem using two datasets. Four deep learning models are introduced, combining Bi and Tri-layer hybrids of BERT and DistilBERT with a Bidirectional Gated Recurrent Unit (BiGRU) algorithm. These methods aim to enhance accuracy while examining the impact of hybridizing BERT and DistilBERT layers with BiGRU. The proposed models were evaluated against standalone BERT and DistilBERT approaches. Among them, the TriDistilBERT with BiGRU architecture achieved the highest accuracy, delivering 91.6% for the WASSA-17 dataset and 99.6% for the BBC dataset.

Keywords: Artificial Intelligence, Intelligent Systems, Machine Learning System, Machine Learning Application, Deep Learning, BERT Model, DistilBERT, BiGRU, Text Classification, Sentiment Classification

Introduction

Because of its outstanding effectiveness in solving complicated issues, the application of Artificial Intelligence (AI) techniques to any real-life sector has grown dramatically in recent years. Natural language processing's crucial role of text categorization has consistently gained attention from academics over the last decade as a result of the potentiality of both classic ML and DL methods (Singh *et al.*, 2022).

Digital technology advancements have considerably contributed to the tremendous amount of information on the internet, particularly text data. Text data is freely accessible via academic publications, news articles, government or organisation files, emails, social media postings and conversations, etc. The majority of this widely accessible text data is unstructured, complex to handle, and requires more storage to evaluate (Adikari *et al.*, 2021; Xu *et al.*, 2024). Text mining, a new field, is used to handle and evaluate massive amounts of unstructured text data (Hearst, 1999). The most crucial component of evaluating a large volume of text data is classifying it. As a result, we focus on text classification, one of the most commonly used approaches in text mining (Guo *et al.*, 2022; Kushwaha *et al.*, 2021).

Text classification is a core issue in natural language processing, and it has numerous solutions such as paper classification, news filtering, sentiment analysis, etc. Many algorithms, particularly DL models, have had tremendous success in text categorization, including recurrent neural networks (Liu *et al.*, 2016; Wang *et al.*, 2018), Convolutional Neural Networks (CNN) (Kim, 2014), and BERT models (Kenton and Toutanova, 2019; Sanh *et al.*, 2019). DL models typically require large amounts of high-quality training data, which may not always be available in real-world scenarios.

Related Work

Yin and Sun (2022) proposed Multi view Clustering with Incomplete Cosine Similarity (IMCCS), a multiview clustering approach that extracts the incomplete multiview data's cosine similarity straight from the original space. The goal of IMCCS is to integrate multiplicative matrices with manifold structure preservation based on cosine similarity. To address the IMCCS optimization issue, using the multiplicative update rule and gradient descent is used. Results of experiments on BBC benchmark datasets. This results in an accuracy of 0.8553. Assuming the hyper parameter can be specified theoretically, the IMCCS method will be more feasible.

In text categorization, Singh *et al.* (2022) decreased the dimensional feature space. The research describes an improved text classification method that uses a Bag-of-

Words encoding model with term frequency inverse document frequency (tf-idf) and the 'GloVe' word embedding methodology to locate words with comparable semantic meaning. Experiment findings on the BBC datasets and others show that the suggested method outperforms existing dimension reduction algorithms in categorization. Using the removal of Redundant Reature (rRF) with a performance analysis accuracy of 96.18 percent. Many factors influence classification performance, including the selection of classification algorithms, the use of various textual representation methods, feature selection approaches, and dimension reduction strategies.

Guo *et al.* (2022) suggested establishing the relationships between the text's terms and their category, as seen from the viewpoint of semantic similarity and statistical correlation and then using them to split the words into roles with distinct capabilities for text classification. Based on these word roles, they introduced Selective Text Augmentation (STA), an augmentation technique in which distinct text-editing procedures selectively affect words with certain roles. With a training data size of 1000, the suggested approach, STA-global, achieves an accuracy of 98.31%. When investigating the role of super-pixels, the method can be expanded to image categorization.

In order to increase the BBC news dataset, Ugwuoke *et al.* (2023) investigated the potential of using WordNet, a semantic lexical data source. The news categorization model in the study was created using the LSTM. Several preprocessing approaches were used, including the user-defined word tagging method and the elimination of stop words. In comparison, an LSTM model without any technique for data augmentation gives 90% accuracy, while the suggested LSTM with a WordNet model achieves an accuracy of 95%. To test the effectiveness of the LSTM with and without the data augmentation technique, they intended to make use of other DL algorithms including BERT with attention mechanism. Additionally, Wikipedia data sources might be more effective at handling compound words or phrases in a particular dataset; altogether, related synonymous words with contextual meaning are anticipated to be produced.

Talaat (2023) used RoBERTa and DistilBERT hybridised with BiGRU and BiLSTM to improve sentiment classification for three datasets and discovered that BiGRU performed best with the emoji dataset. The first collaborative effort to determining the level of emotion experienced by a tweet's speaker was offered by Mohammad and Bravo-Marquez (2017). shared task on emotion intensity for WASSA-2017. Using a method known as (BWS) best worst scaling, they built the first tweet datasets labelled for the intensity of joy, fear, anger, and sadness. They demonstrated how the annotations provide accurate, fine-grained intensity scores and rankings based on the intensity of tweets. In order to demonstrate that affect lexicons, particularly those with high word-emotion association scores, are helpful in assessing emotion intensity, they developed a regression system and carried out tests. The shared task and the emotion intensity dataset are advancing our knowledge of how people express powerful emotions through language. Results produced on the test sets utilising different features, both separately and together. For the WASSA-2017 dataset, the correlations for all emotions with the Word Embedding (WE) and all lexicons (L) (WE + L) result in a macro-average of 66%.

Jacobson *et al.* (2021) studied the association between political polarisation, emotions conveyed in textual posts, and the level of engagement obtained by those messages on Reddit. The dataset utilised for analysis was taken from the two partisan subreddits' yearly top posts, and they used established measures to evaluate polarisation, emotion intensity, and engagement.

They created a naive bayes baseline for evaluating polarisation, as well as two BERT-based models trained to measure polarisation and the intensity of various emotions represented in a post more correctly. They contrasted the breakdown of emotions expressed in postings, the scores assigned to posts by polarisation models, and the amount of engagement posts received.

They trained all neural networks for emotional prediction using the WASSA'17 dataset. The classification accuracy of the Bert models' training outcomes is 85%. They trained all neural networks for emotional prediction using the WASSA'17 dataset. The classification accuracy of the Bert models' training outcomes is 85%.

To address the text classification challenge, Jini and Indra (2021) presented a unique hybrid-based pre-processing technique. The WASSA'17 huge text dataset is first sent into the stage of preprocessing, which employs a variety of hybrid algorithms for removing noisy material. The suggested technique has been verified using three standard classifiers: ANN, PNN, and k-NN. Finally, the performance measures are analysed to demonstrate the efficacy of the suggested methodology. Jini demonstrated a computer-aided system for the recognition of text-based emotional data. The WASSA'17 big text data is initially fed and processed using hybrid pre-processing approaches. The PCA model is then used to extract and reduce its relevant features. Finally, the collected characteristics are sent into the classifier algorithm, which is used to evaluate the system model. The analysis was performed with and without pre-processing, and the results are provided to show how effective the suggested methodology. Before preprocessing, the accuracies of three standard classifier methods, namely PNN, ANN, and k-NN, were 65.6, 58.4, and 67.5, respectively, and 67.8, 61.4, and 71.8, respectively, after preprocessing. The proposed methodology's quality was increased while noisy content from a larger dataset was reduced.

Bharti *et al.* (2022) introduced a hybrid approach combining machine learning and deep learning techniques for emotion recognition in text using Wassa-17 Dataset. The authors used a combination of CNN and Bi-GRU with a support vector machine (SVM) classifier to improve emotion detection accuracy. The model was evaluated using multiple datasets, including sentences, tweets, and dialogs, achieving an accuracy of 80.11%. The study highlights the challenges and effectiveness of emotion detection from unstructured text using deep learning and hybrid methods.

Karaman *et al.* (2023) proposed a comprehensive analysis of SVM, LSTM, and CNN-RNN models for text classification using the BBC News dataset. The study explored various machine learning and deep learning methods, including a hybrid model combining CNN and RNN for improved performance. The results demonstrated the hybrid model's accuracy of 96.0%, with comparative analysis showing the superiority of SVM in achieving 96.0% accuracy. The study provides insights into the efficacy of hybrid models in news text classification.

The proposed method comprises three main components: (Bi and Tri) BERT/DistilBERT and BiGRU layers. BERT and DistilBERT models are employed to generate dynamic contextual word embeddings, enhancing the representation of short-text features. The BiGRU layer captures and learns sequential dependencies within sentences, allowing the model to better understand contextual relationships. Additionally, variations in the number of BERT and DistilBERT layers are explored to optimize efficiency and improve overall model performance.

This study makes several key contributions. First, it introduces four novel hybrid deep learning models for text classification, evaluated using two benchmark datasets. Two of these models, BiBERT and TriBERT, combine BERT with BiGRU layers, while the other two, BiDistilBERT and TriDistilBERT, integrate DistilBERT with BiGRU layers. These hybrid architectures significantly enhance the model's capacity to extract and integrate rich contextual information from text data.

Second, the BiGRU network is used during fine-tuning to improve contextual feature extraction and strengthen classification accuracy. This combination of transformer-based embeddings with a recurrent architecture provides a more comprehensive understanding of sequential text patterns.

Finally, the four proposed models are compared with baseline BERT and DistilBERT methods, demonstrating their superior performance and robustness across datasets. The paper concludes with an overview of experimental findings and suggestions for future research directions.

Materials and Methods

Datasets

To do a multilabel text classification, we used two sets of data that were freely available on the Kaggle site. Each dataset is trained and tested individually.

We divide all datasets into two columns: text and label. Text for tweet text and a label for text representation.

1. WASSA-17 dataset (Mohammad and Bravo-Marquez, 2017): It is the first dataset where systems can determine the level of emotions in tweets automatically. with 10591 rows and 4 classes for: class 0 for 'anger' (2545), class 1 for ‘fear’ (3357), class 2 for ‘joy’ (2409), and class 3 for ‘sadness’ (2280) (Wassa, 2024)
2. BBC dataset (Greene and Cunningham, 2006): It is made up of 2225 papers taken from the BBC news website, with classes: 0 for 'business' (510), 1 for ‘entertainment’ (386), 2 for ‘politics’ (417), 3 for ‘sport’ (511), and 4 for ‘tech’ (401) (BBC News, 2024)

Proposed Models

The primary goal of the article is to categorise the text as business, entertainment, politics, sport, and tech for the BBC dataset, and classify the text as anger, fear, joy, and sadness for the WASSA-17 dataset.

In the data preprocessing phase, we first performed text normalization by removing stop words, special characters, and numeric values to minimize noise and focus on the core, meaningful content. For tokenization, we utilized the tokenizers from the Hugging Face Transformers library, specifically the bert-base-uncased and distilbert-base-uncased models. Lemmatization was carried out using the SpaCy toolkit, ensuring that words were reduced to their base forms while preserving their meaning and eliminating inflectional variants. In terms of outlier handling, any samples containing more than 512 tokens were truncated, and those with fewer than 5 tokens were removed to maintain the quality of the dataset. Finally, for embedding initialization, no additional embeddings were used, relying solely on BERT embeddings for representation.

The new models are series combinations of BERT/DistelBERT models to make (Bi and Tri) layers with BiGRU shown in Figures 1 and 2.

The four models used in detail are as follows:

- BiBERT: BiBERT-BiGRU
- BiDistelBERT: BiDistelBERT-BiGRU
- TriBERT: TriBERT-BiGRU
- TriDistilBERT: TriDistilBERT-BiGRU

The following changes are made to the parameters of the four hybrid approaches stacked with BiGRUs shown

DOI: 10.3844/jcssp.2025.1983.1992

in Figs. 1 and 2:

BiBERT-BiGRU (BiBERT) in Figure 1 where:

‘W’ = Encoder,

‘N’ = 12

‘Z’ = BERT,

‘M’ = 3

’G’ = ‘GRU’,

BiDistelBERT-BiGRU (BiDistelBERT) in Figure 1 where:

‘W’ = Transformer Layer,

‘N’ = 6

‘Z’ = DistelBERT,

‘M’ = 3

’G’ = ‘GRU’,

TriBERT-BiGRU (TriBERT) in Figure 2 where:

‘W’ = Encoder,

‘N’ = 12

‘Z’ = BERT,

‘M’ = 3

’G’ = ‘GRU’,

TriDistilBERT-BiGRU (TriDistilBERT) in Figure 2 where:

‘W’ = Transformer Layer,

‘N’ = 6

‘Z’ = DistelBERT,

‘M’ = 3

’G’ = ‘GRU’,

C = 4 for the WASSA-17 dataset and C = 5 for the BBC dataset.

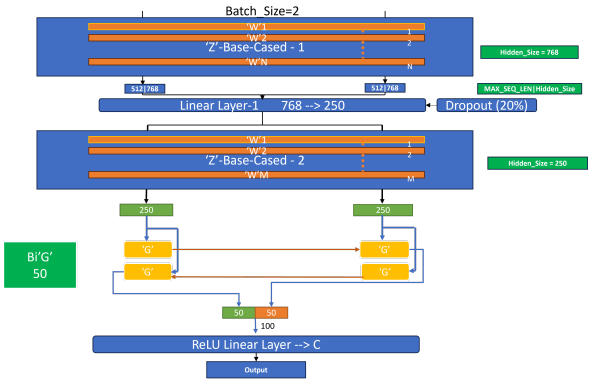


Fig. 1: The BiBERT and BiDistilBERT models stacked with BiGRU layers to capture sequential dependencies in text. Each model includes two BERT/DistilBERT layers followed by a BiGRU layer, with specified parameters for each layer

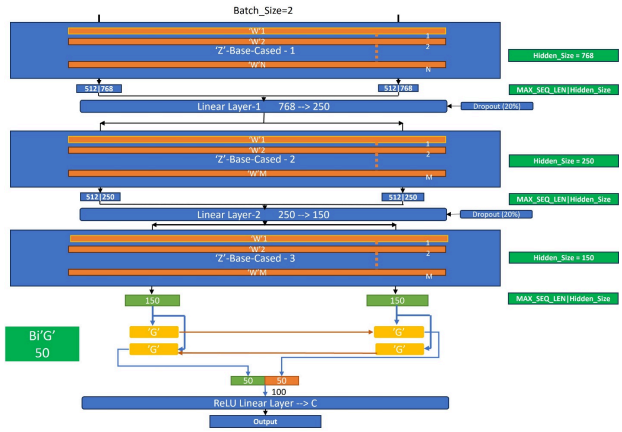


Fig. 2: The TriBERT and TriDistilBERT models with three stacked BERT/DistilBERT layers integrated with a BiGRU layer to extract more complex features from the input text. The models are designed to improve sequential feature learning and accuracy

Bi‘Z’ Classifier Architecture

The model consists of two ‘Z’ models: (‘Z’-base-cased1) and (‘Z’-base-cased2) for text classification. The first ‘Z’ model (‘Z’-base-cased1) is set up with the pre-trained weights. A dropout layer is added following the first linear layer. The second ‘Z’ model (‘Z’-base-cased2) is initialized with a custom (’Z’Config) object that sets the `hidden_size` to 250. In (‘Z’-base-cased2), three specific layers from the ‘Z’ encoder (‘W1’, ‘W2’, ‘WM`) are extracted. The output is (2*250) features. We build a (Bi’G’) layer and feed these 500 features through it along with the 50 hidden features. As output from this (Bi’G’) layer, we obtained 100 features as an output. Next, we apply the ReLU activation function and convert the output of (Bi’G’) layer into a linear layer with an output of ‘C’ classes. The classifier also includes a linear layer: ‘Linear Layer-1’ with input and output dimensions of 768->250.

Tri‘Z’ Classifier Architecture

The model consists of three ‘Z’ models for text classification. The first ‘Z’ model, (‘Z’-base-cased1), is set up with the pre-trained weights. The second ‘Z’ model, (‘Z’-base-cased2), is initialized with a custom (’Z’Config) object that sets the hidden_size to 250. The third ‘Z’ model, (‘Z’-base-cased3), is initialized with a custom (’Z’Config) object that sets the hidden_size to 150. In (‘Z’-base-cased2) and (‘Z’-base-cased3), three specific layers from the ‘Z’ encoder (‘W1’, ‘W2’, ‘WM`) are extracted. The output of (‘Z’-base-cased3) is (2*150) features. We build a (Bi’G’) layer and feed these 300 features through it along with the 50 hidden features. As output from this (Bi’G’) layer, we obtained 100 features as an output. Next, we apply the ReLU activation function and convert the output of (Bi’G’) layer into a linear layer with an output of ‘C’ classes. The classifier also includes two linear layers: ‘Linear Layer-1’, and

'Linear Layer-2' with input and output dimensions of 768->250, 250->150 respectively. A dropout layer is applied after the first and second linear layers.

Figure 1 illustrates the architecture of the BiBERT and BiDistilBERT models integrated with a Bidirectional Gated Recurrent Unit (BiGRU) layer.

- BiBERT: We utilize the BERT model with two stacked layers (denoted as 'Z'-Base-Cased-1 and 'Z'-Base-Cased-2). Each layer in BERT is responsible for processing the input text to extract contextual word embeddings, which are then passed through the BiGRU layer for capturing sequential dependencies in the text.
- BiDistilBERT: Similar to BiBERT, the BiDistilBERT model is a lightweight variant of BERT that retains most of the model's capabilities with fewer parameters, making it more computationally efficient. It also incorporates two DistilBERT layers for extracting features and a BiGRU for sequential feature learning.
- BiGRU Layer: The BiGRU layer processes the embeddings from BERT/DistilBERT, enhancing the model's ability to capture bidirectional context and relationships between tokens in the sequence.
- Parameter Values: We also specify the hidden sizes for each layer, such as 768 for the BERT layers and 250 for the BiGRU output. Additionally, we introduce dropout (20%) after the linear transformation layers to regularize the model.

Figure 2 depicts a similar architecture as in Figure 1 but with three stacked layers of BERT/DistilBERT.

- TriBERT: The TriBERT model extends the BiBERT architecture by adding an additional BERT layer, making it capable of capturing more intricate features from the input text. Each additional layer contributes to deeper contextual understanding.
- TriDistilBERT: Similarly, TriDistilBERT incorporates three DistilBERT layers, increasing its capacity for feature extraction while maintaining computational efficiency.
- BiGRU Layer: The BiGRU layer in TriBERT/TriDistilBERT captures bidirectional dependencies in the sequential data, enhancing the overall model performance for text classification tasks.
- Parameter Values: In the TriBERT model, the hidden sizes are 768 for the first layer, 250 for the second, and 150 for the third. Dropout layers are also applied after the first two linear layers for regularization.

In both architectures, the final output is passed through a ReLU activation function and a linear layer to classify the input into one of the predefined classes.

The output of four classes: 0,1,2,3 (anger, fear, joy, and sadness) for the WASSA-17 dataset, and an output of five classes: 0,1,2,3,4 (business, entertainment, politics, sport, and tech) for the BBC dataset.

The models are set up with the following parameters: AdamW is the Optimizer, 1e-6 is the learning rate, 10 is the number of epochs, and 2 is the batch size. We set the loss to 'Cross Entropy Loss' and trained the model.

The models were configured with AdamW as the optimizer, a learning rate of 1e-6, 10 epochs, and a batch size of 2. The loss function used was Cross Entropy Loss, and the model was trained accordingly. To fine-tune these parameters, we performed a grid search over a predefined set of hyperparameters, including learning rates [1e-5, 1e-6, 1e-7], batch sizes [2, 4], and the number of epochs [5 to 15]. The evaluation was based on the validation F1-score across five-fold cross-validation. The optimal configuration, determined through this search, was a learning rate of 1e-6, a batch size of 2, and 10 epochs. The choice of a small batch size was due to GPU memory constraints, while still ensuring stable training. Models generally converged well around the 10-epoch mark, and higher learning rates led to instability and overfitting in transformer-based models.

Experiments and Outcomes

We used two datasets in this study to train the classifier, validate, and test the system. The data was divided into three categories: training (80%), testing (10%), and validation (10%). On Kaggle, the task was accomplished with a 2.3 GHz Intel (R) Xeon (R) CPU 2.20GHz, 16 GB of RAM, and Nvidia P100 GPU.

Results

We ran the six models on two datasets. Table 1 displays Performance Metrics of all the findings.

Table 1: Performance Metrics of BBC and Wassa-17 datasets

	Wassa				BBC			
	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
BERT	0.906	0.903	0.910	0.906	0.978	0.976	0.977	0.978
DistilBERT	0.907	0.904	0.915	0.908	0.987	0.987	0.986	0.986
BERT+CNN	0.909	0.906	0.912	0.909	0.982	0.980	0.981	0.981
BERT+LSTM	0.911	0.909	0.914	0.911	0.984	0.983	0.983	0.983
BERT+BiLSTM	0.912	0.910	0.915	0.912	0.986	0.985	0.985	0.985
BiBERT	0.913	0.912	0.915	0.913	0.991	0.989	0.990	0.989
BiDistlBERT	0.915	0.913	0.917	0.915	0.991	0.991	0.990	0.990
TriBERT	0.915	0.912	0.920	0.916	0.996	0.994	0.995	0.994
TriDistilBert	0.916	0.913	0.921	0.916	0.996	0.994	0.995	0.994

DOI: 10.3844/jcssp.2025.1983.1992

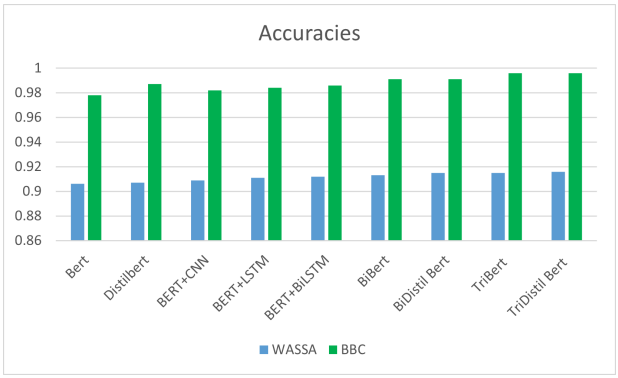


Fig. 3: BBC and WASSA-117 accuracy comparison between models

Figure 3 depicts a comparison graph of nine models. The four best approaches' ROC curves (BiBERT, BiDistlBERT, TriBERT, and TriDistilBert) for each dataset are shown in Figures 4 and 5. While the confusion matrix of the best four methods for each dataset is shown in Figures 6 and 7. Also the classification report of the best four methods for each dataset is shown in Figures 8 and 9. Finally Performance Metrics of the six methods are shown in details in Figures 10 and 11.

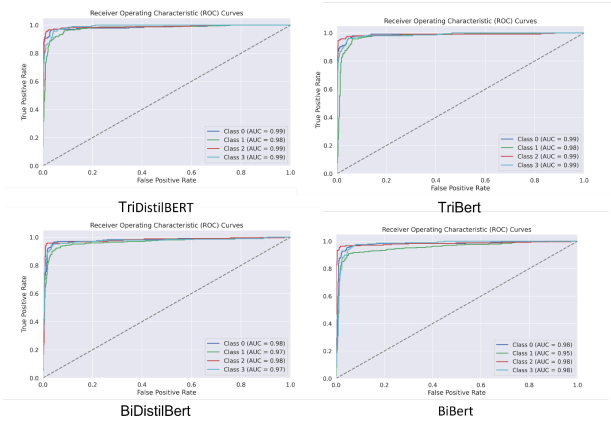


Fig. 4: ROC curves for the WASSA-17 dataset's highest accuracy

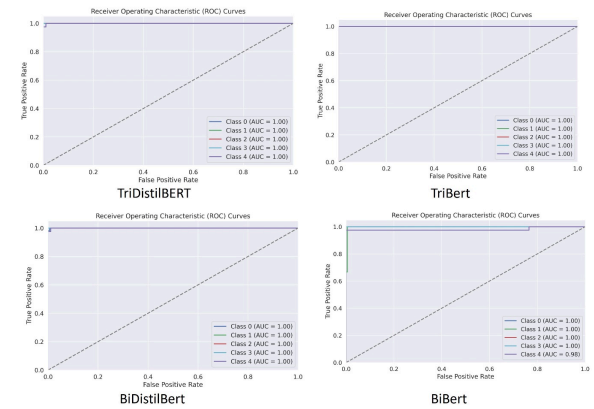


Fig. 5: ROC curves for the BBC dataset's highest accuracy

For Figure 6 there are the most misclassifications in the confusion matrix. Specifically, fear was misclassified as anger, and fear were misclassified as sadness. This suggests that the model had difficulty distinguishing fear from anger and sadness, likely due to the similarity in emotional content between these categories. These misclassifications likely arise from the nuanced nature of these emotions, which may appear similar in the context of the dataset's textual data.

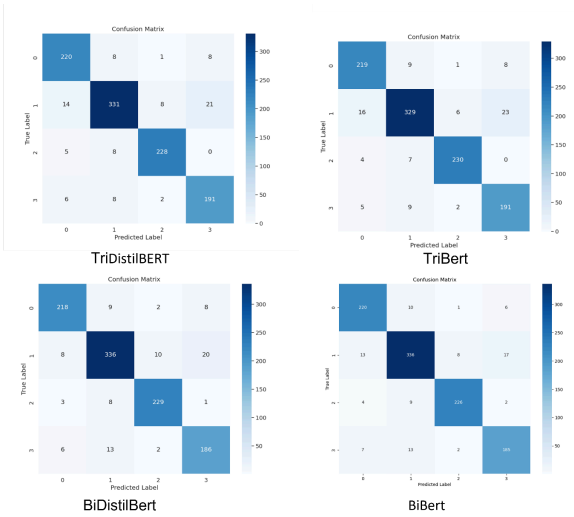


Fig. 6: Confusion matrix of highest Accuracies of Wassa-17 dataset

For Figure 7 there are few misclassifications here, joy is misclassified as anger. However, this is not as significant as the misclassifications seen with the fear class.

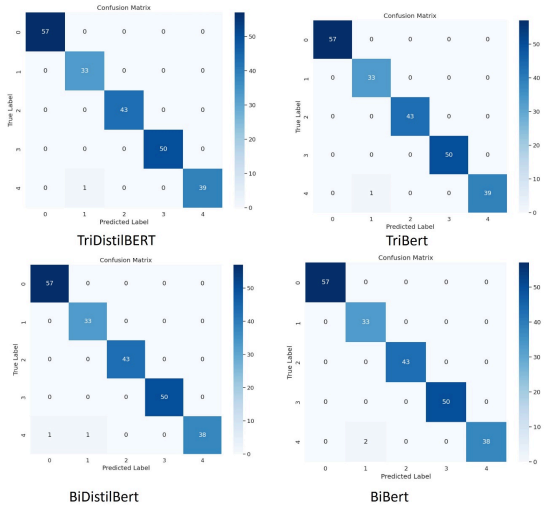


Fig. 7: Confusion matrix of highest Accuracies of BBC dataset

It is interesting to notice that the values of AUC-ROC of Figure 5 for BBC dataset is nearly 1 for the best four methods. But these methods have different accuracies which are less than 1. Because AUC=1 indicates that the model has excellent discriminative power. but it doesn't

DOI: 10.3844/jcssp.2025.1983.1992

necessarily imply that class-specific metrics (accuracy, precision, recall, or F1-score) for each class will be 1. This is especially true when there are misclassifications within those classes.

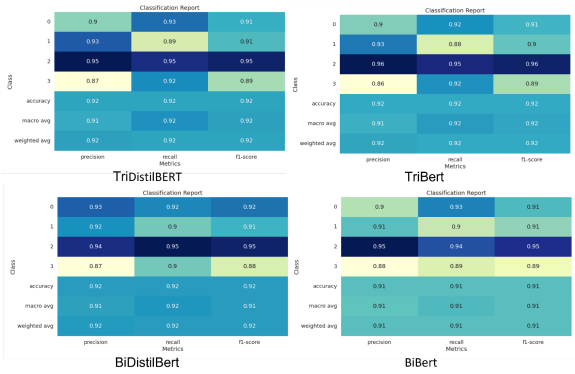


Fig. 8: Classification report of best Accuracies of Wassa-17 dataset

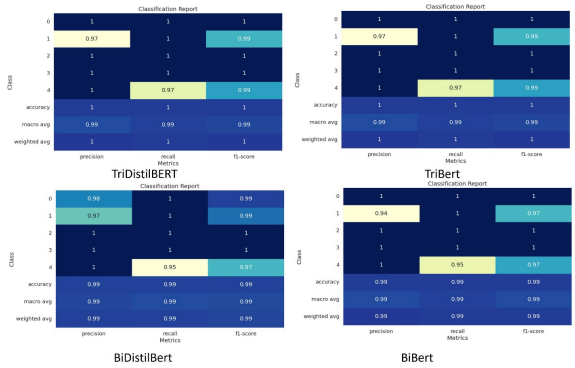


Fig. 9: Classification report of best Accuracies of BBC dataset

We improved accuracy by hybridising layers of BERT/DistlBERT in series and stacking models with BiGRU.

For the WASSA-17 dataset in Table 1 and Figure 3, the most accurate way is TriDistilBert with 0.916. Equal accuracies of 0.915 for BiDistlBERT, and TriBERT and the least accuracy of 0.913 for BiBERT.

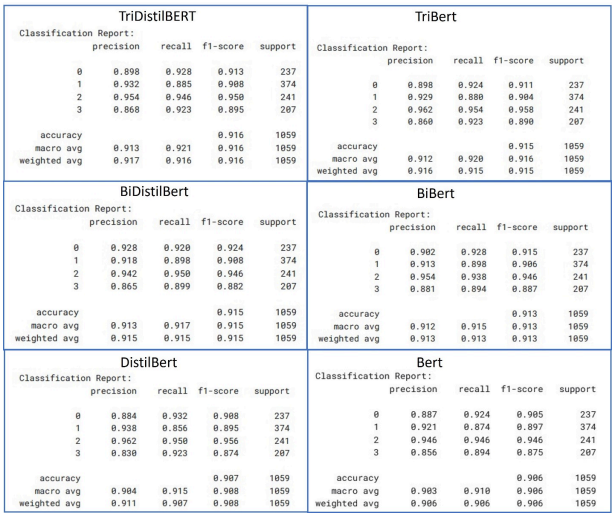
For the BBC dataset, TriDistilBert and TriBERT have the same accuracy of 0.996 followed by BiDistlBERT, and BiBERT with an equal accuracy of 0.991.

TriDistilBert and TriBERT models of BBC dataset are performing equally well on the given dataset. Because they have the same performance metrics.

BiDistlBERT, and BiBERT models of BBC dataset have equal accuracy of 0.991 but BiDistlBERT is higher than BiBERT in Precision 0.991 and F1-score 0.990 which means BiDistlBERT aims to minimize false positives. This is reflected in its higher precision score. Also BiDistlBERT has a higher F1-score, so it performs slightly better than BiBERT.

We run Bi and Tri BERT/DistilBERT layers on the two datasets to get higher accuracies.

We proved that hybridizing layers of BERT and DistelBERT works better than a single BERT and DistelBERT model.



TriDistilBERT

Classification Report:

	precision	recall	f1-score	support
0	0.898	0.928	0.913	237
1	0.932	0.885	0.908	374
2	0.954	0.946	0.950	241
3	0.868	0.923	0.895	207
accuracy			0.916	1059
macro avg	0.913	0.921	0.916	1059
weighted avg	0.917	0.916	0.916	1059

TriBert

Classification Report:

	precision	recall	f1-score	support
0	0.898	0.924	0.911	237
1	0.929	0.880	0.904	374
2	0.962	0.954	0.958	241
3	0.860	0.923	0.890	207
accuracy			0.915	1059
macro avg	0.912	0.920	0.916	1059
weighted avg	0.916	0.915	0.915	1059

BiDistilBert

Classification Report:

	precision	recall	f1-score	support
0	0.928	0.920	0.924	237
1	0.918	0.898	0.908	374
2	0.942	0.950	0.946	241
3	0.865	0.899	0.882	207
accuracy			0.915	1059
macro avg	0.913	0.917	0.915	1059
weighted avg	0.915	0.915	0.915	1059

BiBert

Classification Report:

	precision	recall	f1-score	support
0	0.902	0.928	0.915	237
1	0.913	0.898	0.906	374
2	0.954	0.938	0.946	241
3	0.881	0.894	0.887	207
accuracy			0.913	1059
macro avg	0.912	0.915	0.913	1059
weighted avg	0.913	0.913	0.913	1059

DistilBert

Classification Report:

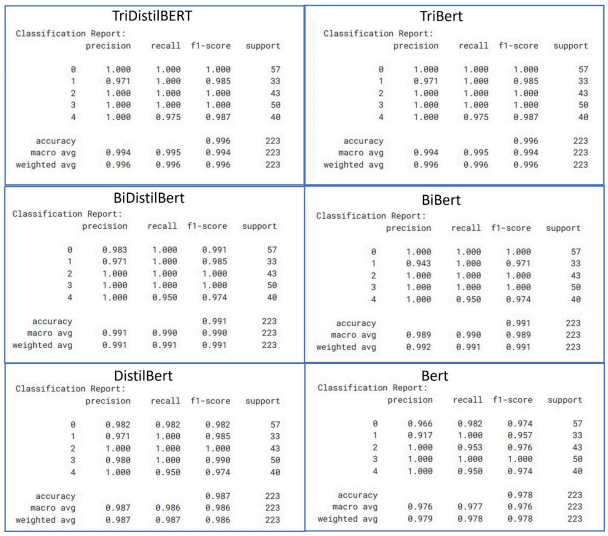
	precision	recall	f1-score	support
0	0.884	0.932	0.908	237
1	0.938	0.856	0.895	374
2	0.962	0.950	0.956	241
3	0.830	0.923	0.874	207
accuracy			0.907	1059
macro avg	0.904	0.915	0.908	1059
weighted avg	0.911	0.907	0.908	1059

Bert

Classification Report:

	precision	recall	f1-score	support
0	0.887	0.924	0.905	237
1	0.921	0.874	0.897	374
2	0.946	0.946	0.946	241
3	0.856	0.894	0.875	207
accuracy			0.906	1059
macro avg	0.903	0.910	0.906	1059
weighted avg	0.906	0.906	0.906	1059

Fig. 10: Performance Metrics values of Wassa-17 datasets



TriDistilBERT

Classification Report:

	precision	recall	f1-score	support
0	1.000	1.000	1.000	57
1	0.971	1.000	0.985	33
2	1.000	1.000	1.000	43
3	1.000	1.000	1.000	50
4	1.000	0.975	0.987	40
accuracy			0.996	223
macro avg	0.994	0.995	0.994	223
weighted avg	0.996	0.996	0.996	223

TriBert

Classification Report:

	precision	recall	f1-score	support
0	1.000	1.000	1.000	57
1	0.971	1.000	0.985	33
2	1.000	1.000	1.000	43
3	1.000	1.000	1.000	50
4	1.000	0.975	0.987	40
accuracy			0.996	223
macro avg	0.994	0.995	0.994	223
weighted avg	0.996	0.996	0.996	223

BiDistilBert

Classification Report:

	precision	recall	f1-score	support
0	0.983	1.000	0.991	57
1	0.971	1.000	0.985	33
2	1.000	1.000	1.000	43
3	1.000	1.000	1.000	50
4	1.000	0.950	0.974	40
accuracy			0.991	223
macro avg	0.991	0.990	0.990	223
weighted avg	0.991	0.991	0.991	223

BiBert

Classification Report:

	precision	recall	f1-score	support
0	1.000	1.000	1.000	57
1	0.943	1.000	0.971	33
2	1.000	1.000	1.000	43
3	1.000	1.000	1.000	50
4	1.000	0.950	0.974	40
accuracy			0.991	223
macro avg	0.989	0.990	0.989	223
weighted avg	0.992	0.991	0.991	223

DistilBert

Classification Report:

	precision	recall	f1-score	support
0	0.982	0.982	0.982	57
1	0.971	1.000	0.985	33
2	1.000	1.000	1.000	43
3	0.980	1.000	0.990	50
4	1.000	0.950	0.974	40
accuracy			0.987	223
macro avg	0.987	0.986	0.986	223
weighted avg	0.987	0.987	0.986	223

Bert

Classification Report:

	precision	recall	f1-score	support
0	0.966	0.982	0.974	57
1	0.917	1.000	0.957	33
2	1.000	0.953	0.976	43
3	1.000	1.000	1.000	50
4	1.000	0.950	0.974	40
accuracy			0.978	223
macro avg	0.976	0.977	0.976	223
weighted avg	0.979	0.978	0.978	223

Fig. 11: Performance Metrics values of BBC datasets

In general, for the WASSA datasets, TriDistilBERT is the best model. Because DistilBERT is a smaller and faster version of BERT that requires fewer computational resources and less memory. While BERT has a large number of parameters (around 340 million), DistilBERT has only 66 million parameters. While we didn't see any difference in accuracy in the BBC dataset in Bi and Tri BERT/DistilBERT because the dataset is much smaller.

The smaller size of DistilBERT means that it can be trained faster and requires less computational power, making it a more efficient model.

TriBERT/TriDistilBERT is a variant of BERT/DistilBERT that uses a three-layer transformer architecture instead of the two-layer transformer architecture used by BiBERT/BiDistilBERT. The additional layers in TriBERT/TriDistilBERT allow it to

capture more complex patterns in the input data, making it possible to achieve higher accuracy on some tasks than BiBERT/BiDistilBERT.

The BBC dataset has perfect classification accuracy (99.6%) suggests that the dataset may be linearly separable or potentially affected by data leakage. Upon inspection, we confirmed that the dataset is well-defined, balanced, and structured, which played a significant role in facilitating model performance. We applied a strict stratified split to ensure proper separation of training and testing data. The dataset's clean and organized format, coupled with clearly distinguishable classes (business, entertainment, politics, sport, and tech), allowed the model to effectively learn discriminative features, which likely contributed to the high accuracy. Additionally, the small size and clean structure of the dataset, combined with TriDistilBERT's strong contextual embedding capabilities, further enhanced the model's performance.

The computational complexity and runtime efficiency of hybrid models depend on various factors, including the types of base models used, the hybridization approach, and the computational resources available. While hybrid models can lead to improved performance, they often come with increased training and inference time. However, careful model selection, optimization strategies, and techniques like TriDistilBERT's can help mitigate these challenges, ensuring that hybrid models remain computationally feasible for practical applications.

Performance Comparison

A comparison of prior papers' accuracy is offered in Table 2. When comparing accuracies, the TriDistilBERT model stacked with BiGRU is the best model for the two datasets, but it is equal to TriBERT in the BBC dataset, followed by BiDistlBERT which is better in the WASSA dataset but equal to BiBERT in BBC dataset.

DistilBERT is more effective than BERT because it has a structure that is less complicated. This efficiency can be shown in our results.

Table 2: Comparison of our method with other methods

Model	Dataset	Accuracy	Notes
Proposed TriDistilBert	WASSA-17	0.916	
	BBC	0.996	
Proposed TriBERT	WASSA-17	0.915	
	BBC	0.996	
Proposed BiDistlBERT	WASSA-17	0.915	
	BBC	0.991	
Proposed BiBERT	WASSA-17	0.913	
	BBC	0.991	
Mohammad and Bravo-Marquez, 2017	WASSA-17	Macro-Average 0.66	WE + L
Jacobson *et al*., 2021	WASSA-17	0.85	BERT
Jini and Indra, 2021	WASSA-17	Before Preprocessing: KNN= 0.675 ANN=0.584 PNN=0.656 After Preprocessing: KNN= 0.718 ANN=0.614 PNN=0.678	KNN, ANN, PNN
Yin and Sun, 2022	BBC	0.8553	IMCCS
Singh *et al*., 2022	BBC	0.9618	rRF
Guo *et al*., 2022	BBC	0.98.31	STA-global, N=1000
Ugwuoke *et al*., 2023	BBC	0.95	Augmented LSTM+WordNet

Discussion

Our results show that hybrid models combining BiGRU with multiple stacked layers of BERT and DistilBERT outperform baseline deep learning models in text classification tasks.

TriDistilBERT-BiGRU outperformed others for the WASSA-17 dataset, reaching 91.6% accuracy. TriBERT-BiGRU achieved the same top accuracy (99.6%) as TriDistilBERT on the BBC dataset. This confirms that adding a third transformer layer improves feature extraction and DistilBERT, though smaller, performs nearly as well as BERT with fewer resources.

The structured nature of the BBC dataset likely contributes to the nearly perfect classification. Stratified splitting was used to avoid data leakage.

Conclusion

In many fields, text analysis is essential for understanding public writings and tweets and for formulating strategic judgments. This study describes hybridising strategies for creating a DL model for text classification with multiple labels for two datasets.

Hybridizing Bi and Tri BERT/DistelBERT layers stacked with BiGRU in series to increase accuracy. The combination of BIGRU layers with BERT and DistelBERT layers enhances the accuracy.

Four hybrid models are proposed, and TriDistilBERT achieved a 0.9% increase over DistilBERT alone for the

WASSA-17 and BBC datasets. TriBERT achieved a 0.9 and 1.8% respectively increase over BERT alone for the WASSA-17, and BBC datasets repectively. BiDistilBERT achieved a 0.8 and 0.7% respectively increase over DistilBERT alone for the WASSA-17 and BBC datasets. BiBERT achieved a 0.7 and 1.3% respectively increase over BERT alone for the WASSA-17, and BBC Datasets. In General, our models produced the greatest results with 91.6% accuracy for WASSA-17 and 99.6% for BBC datasets. In the future, we would like to continue this project by making other hybrid combinations to see what will be better.

In future work, we aim to enhance the interpretability of our model by integrating SHAP values for feature-level insights and attention heatmaps for visualizing word-level contributions. Additionally, we plan to optimize computational efficiency through strategies like model pruning and parallelization, ensuring better scalability and performance for real-world applications.

Acknowledgment

"My mother, I wish you could see me from the sky and be proud of me." "I love you, my father."

Ethics

This research did not involve human participants or animals, and all datasets were used with proper permissions. The study adheres to ethical research practices, with no conflicts of interest declared.

References

Adikari, A., Burnett, D., Sedera, D., de Silva, D., & Alahakoon, D. (2021). Value co-creation for open innovation: An evidence-based study of the data driven paradigm of social media using machine learning. *International Journal of Information Management Data Insights*, *1*(2), 100022. https://doi.org/10.1016/j.jjimei.2021.100022

BBC News. (2024). *BBC News*. https://www.kaggle.com/datasets/sainijagjit/bbc-dataset.

Bharti, S. K., Varadhaganapathy, S., Gupta, R. K., Shukla, P. K., Bouye, M., Hingaa, S. K., & Mahmoud, A. (2022). Text-Based Emotion Recognition Using Deep Learning Approach. *Computational Intelligence and Neuroscience*, *2022*, 1-8. https://doi.org/10.1155/2022/2645381

Greene, D., & Cunningham, P. (2006). Practical solutions to the problem of diagonal dominance in kernel document clustering. *Proceedings of the 23rd International Conference on Machine Learning*, 377-384. https://doi.org/10.1145/1143844.1143892

Guo, B., Han, S., & Huang, H. (2022). Selective text augmentation with word roles for low-resource text classification. *ArXiv:2209.01560*. https://doi.org/10.48550/arXiv.2209.01560

Hearst, M. A. (1999). Untangling text data mining. *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics, College Park, Maryland*. https://doi.org/10.3115/1034678.1034679

Jacobson, N., Liang, J., & Gao, H. (2021). Political Polarization, Emotions and Engagement on Reddit. *Stanford University, Department of Computer Science*.

Jini, S. S., & Indra, N. C. (2021). Noise Destruction Towards Quality Improvement in Emotion Recognition from Text Using Pre-Processing Modules. *Optical Memory and Neural Networks*, *30*(3), 214-224. https://doi.org/10.3103/s1060992x21030097

Karaman, Y., Akdeniz, F., Savaş, B. K., & Becerikli, Y. (2023). A Comparative Analysis of SVM, LSTM and CNN-RNN Models for the BBC News Classification. *Innovations in Smart Cities Applications Volume 6*, 473-483. https://doi.org/10.1007/978-3-031-26852-6_44

Kenton, J. D., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of NaacL-HLT*, 4171-4186.

Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar. https://doi.org/10.3115/v1/d14-1181

Kushwaha, A. K., Kar, A. K., & Dwivedi, Y. K. (2021). Applications of big data in emerging management disciplines: A literature review using text mining. *International Journal of Information Management Data Insights*, *1*(2), 100017. https://doi.org/10.1016/j.jjimei.2021.100017

Liu, P., Qiu, X., & Huang, X. (2016). Recurrent neural network for text classification with multi-task learning. *ArXiv:1605.05101*. https://doi.org/10.48550/arXiv.1605.05101

Mohammad, S., & Bravo-Marquez, F. (2017). WASSA-2017 Shared Task on Emotion Intensity. *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 34-49. https://doi.org/10.18653/v1/w17-5205

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter. *ArXiv:1910.01108*. https://doi.org/10.48550/arXiv.1910.01108

Singh, Ksh. N., Devi, S. D., Devi, H. M., & Mahanta, A. K. (2022). A novel approach for dimension reduction using word embedding: An enhanced text classification approach. *International Journal of Information Management Data Insights*, *2*(1), 100061. https://doi.org/10.1016/j.jjimei.2022.100061

Talaat, A. S. (2023). Sentiment analysis classification system using hybrid BERT models. *Journal of Big Data*, *10*(1), 110. https://doi.org/10.1186/s40537-023-00781-w

Ugwuoke, U. C., Aminu, E. F., & Ekundayo, A. (2023). Performing Data Augmentation Experiment to Enhance Model Accuracy: A Case Study of BBC News' Data. *Proceedings of International Conference on Information Systems and Emerging Technologies*, 12. https://doi.org/10.2139/ssrn.4333014

Wang, W., Gan, Z., Wang, W., Shen, D., Huang, J., Ping, W., Satheesh, S., & Carin, L. (2018). Topic compositional neural language model. *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, 356-365.

Wassa. (2024). https://www.kaggle.com/datasets/phantomrider/twitter-emotion

Xu, W., Chen, J., Ding, Z., & Wang, J. (2024). Text sentiment analysis and classification based on bidirectional Gated Recurrent Units (GRUs) model. *Applied and Computational Engineering*, *77*(1), 132-137. https://doi.org/10.54254/2755-2721/77/20240670

Yin, J., & Sun, S. (2022). Incomplete multi-view clustering with cosine similarity. *Pattern Recognition*, *123*, 108371. https://doi.org/10.1016/j.patcog.2021.108371