ORIGINAL ARTICLE

Evaluating BERT-based language models for detecting misinformation

Rini Anggrainingsih[1,2] · Ghulam Mubashar Hassan[2] · Amitava Datta[2]

Received: 12 May 2022 / Accepted: 19 February 2025 / Published online: 10 March 2025

Abstract

Online misinformation poses a significant challenge due to its rapid spread and limited supervision. To address this issue, automated rumour detection techniques are essential for countering the negative impact of false information. Previous research primarily focussed on extracting text features, which proved time-consuming and less effective. In this study, we contribute substantially to two domains: rumour detection on Twitter and the evaluation of text embeddings. We thoroughly analyse rumour detection models and compare the quality of text embeddings generated by various fine-tuned BERT-based models. Our findings indicate that our proposed models outperform existing techniques. Notably, when we test these models on combined datasets, we observe significant performance improvements with larger training and testing data sizes. We conclude that carefully considering the dataset, data splitting, and classification techniques is crucial for evaluating solution performance. Additionally, we find that differences in the quality of text embeddings between RoBERTa, BERT, and DistilBERT are insignificant. This challenges existing assumptions and highlights the need for future research to explore these nuances further.

Keywords BERT-based models · Comparative analysis · Rumour detection · Text embeddings

1 Introduction

The rapid growth of internet technology and online social media has revolutionised communication. People can easily access the latest news, share opinions, and interact on various platforms. However, this convenience has also fuelled the spread of false information, including rumours and fake news. The impact can be severe, causing financial losses, public panic, reputational damage, and even risks to human life.

Recently, the spread of rumours and fake news on online media has become a serious issue. Most people find it challenging to distinguish between real information and false rumours on most online platforms. Consequently, there is an urgent need for automated detection technology to identify and flag rumours or fake news on online social media.

Castillo et al. [1] are well-known pioneers in credible information detection research. They used feature engineering and traditional machine-learning techniques to identify rumours on Twitter. Their work established a baseline for subsequent studies. Following their work, other researchers have worked to improve misinformation detection by identifying key features in the context and content of the textual data. Additionally, several other studies compared the performance of various traditional machine-learning algorithms for rumour detection. These include random forest, support vector machine, K-nearest neighbour, naïve Bayes classifier, and logistic regression [2–5].

Ghulam Mubashar Hassan and Amitava Datta have contributed equally to this work.

https://doi.org/10.1007/s00521-025-11101-z

With the success of neural network approaches in classification tasks, more recent studies adopted various neural network techniques to detect rumours and fake news on online media. These approaches include recurrent neural network (RNN) [6–8], convolutional neural network (CNN) [9–13], and long short-term memory (LSTM) network [14–16]. Additionally, researchers have explored innovative models such as the statistical features fusion network [17], spatial–temporal structure neural network [18], and propagation path aggregation network [19], reinforcement learning [20], entity recognition and sentence configuration [21], and graph-based neural networks [22–26]. Moreover, other studies combined traditional and deep learning algorithms [27, 28] to enhance detection performance. These techniques show the evolving research focussed on tackling the challenges of identifying and fighting misinformation on online platforms.

There are two primary classification approaches in the realm of rumour detection: context-based and content-based methods [1–3, 5]. Context-based features encompass information linked to tweets, including details about the user and their network. In contrast, content-based features involve extracting information from a tweet's text, such as linguistic elements and semantic cues. Most approaches, whether using context-based or content-based strategies, extract features from tweets and convert them into numerical data. These are then classified using traditional machine-learning models [3–5, 29] or deep neural network-based models [8, 12, 15].

Recent studies have achieved advanced results in detecting rumours on Twitter by using graph-based neural networks that explore both content- and context-based features. These approaches create high-level representations from propagation paths, trees, or networks to identify rumours [23]. However, extracting network information from tweets on Twitter is complex and time-consuming. Additionally, Twitter's policies limit users to sharing only the tweet ID, requiring the use of Twitter application programming interfaces (APIs) to retrieve all tweet attributes, like text, user IDs, and retweet counts. This process involves mining multiple layers of data to obtain comprehensive information. For example, in the initial stage, the tweet's text, user IDs, and retweet count are retrieved. Then, a second layer is mined to obtain the poster's information or user details. Similarly, further layers must be mined to extract retweet text, images, or videos. As a result, gathering all this information is a time-intensive process.

Moreover, Twitter has limits on data downloading. If the number of retrieved tweets exceeds the limit, data streaming stops. Twitter also restricts data flow to a 15-minute window. Additionally, past tweets are only available for up to three weeks unless a user subscribes to an enterprise API or requests access to a full-search database, which can be costly.

All published datasets include only basic context and content details about tweets. To get more information, APIs are needed to extract additional data from Twitter. However, since users can delete their tweets at any time, many tweets may no longer exist, making it impossible to retrieve all details about them. This means the network and propagation data do not fully reflect conditions when a rumour spreads as breaking news. Additionally, many users keep strict privacy settings, restricting access to their information. Given these limitations, this study focuses on content-based rumour detection models using only text through text embedding.

The advent of text embeddings, like Word2Vec [30], GloVe [31], and FastText [32], represented a pivotal moment in modern language models. These text embeddings transform text into numerical vectors, enabling words that convey similar meanings to possess comparable numerical representations. Another notable advancement in the field of natural language processing (NLP) is the introduction of transformers [33], which necessitate training on substantially larger and more varied collections of text.

Recently, the emergence of bidirectional encoder representation from transformer (BERT) as a pre-trained language model (LM) in 2018 by Devlin et al. [34] has significantly advanced the field of natural language processing (NLP). BERT's ability to comprehend a word's contextual meaning by considering both left- and right-side words has proven to be highly effective. Moreover, it can be fine-tuned to tackle various NLP-related tasks, including text classification [35–39]. BERT's exceptional performance has made it a prominent research trend, inspiring a new era and encouraging researchers to refine BERT for specific goals. For example, RoBERTa (robustly optimised BERT approach) [40] is a pre-trained language model with 24 hidden layers and 355 million parameters, based on BERT$_{\mathrm{LARGE}}$, but trained on a larger 160GB dataset with bigger batch sizes and more iterations. Another example is BERTweet [41], a pre-training model that considers unique Twitter characteristics such as hashtags, mentions, and URLs to understand the semantics of tweets and specific linguistic features unique to Twitter. Similarly, CT-BERT (COVID Twitter BERT) [42] is a pre-trained language model specifically trained on a large corpus of COVID-19-related tweets. Although larger BERT models perform well, their high memory and computational demands limit their use in resource-constrained settings. To address this, researchers have developed less complex alternatives that maintain strong performance, such as DistillBERT [43], TinyBERT [44], ALBERT [45], and MobileBERT [46].

The impressive classification abilities and growing variety of BERT models, despite their high resource demands, drive this study. This raises a key question in rumour detection on Twitter: Do we really need complex models, and which BERT variant provides the best text embeddings for rumour detection on Twitter? For this reason, this study compares the quality of text embeddings from three BERT-based models of different sizes (large, medium, and small): RoBERTa, BERT, and DistilBERT. Each model was fine-tuned on labelled datasets, generating vectors that were used to train neural network classifiers for rumour detection. The models' performance was evaluated on Twitter datasets, including Pheme dataset [15], Twitter 15 dataset, and Twitter 16 dataset [6].

In this study, we contribute substantially to two critical areas: rumour detection on social media, specifically Twitter, and the quality assessment of text embeddings generated from various-sized fine-tuned BERT-based language models. Our research aims to offer valuable insights into these domains, enhancing our understanding and potentially guiding future advancements in natural language processing (NLP). We achieve this through the following key details.

- We present a new model that improves rumour detection on Twitter by using text embeddings from fine-tuned BERT-based models combined with neural networks.
- We conduct a thorough evaluation of three classifier models and BERT-based models for rumour detection across multiple datasets. This includes detailed comparisons on training, validation, and testing sets, with results on various parameter settings, fine-tuning strategies, confusion matrices, and accuracy graphs.
- We establish a new benchmark testing protocol that uniformly allocates datasets for training, validation, and testing, ensuring a fair and robust evaluation framework.
- We demonstrate the superior performance of our proposed model compared to recent state-of-the-art (SOTA) techniques in rumour and fake news detection.

The subsequent sections of this paper are structured as follows: Sects. 2 and 3 provide a comprehensive review of related work on rumour detection and elaborate on the proposed model, which harnesses pre-trained language models and neural networks for rumour and fake news detection, respectively. Section 4 delves into the experimental procedures and discusses the model's performance. Finally, Sect. 5 summarises the key findings and contributions of our study.

2 Related work

This section explains the spreading of rumours and fake news on online social media. Recent studies on rumour and fake news detection techniques are then categorised and discussed systematically in different subsections.

2.1 Rumour and fake news detection

Although studies may offer different definitions, rumour, and fake news are often used interchangeably. A rumour is a story that appears credible but is difficult to verify due to some information remaining unverified [47]. On the other hand, fake news refers to false information deliberately spread to gain biased public opinion on specific issues [48]. The lack of supervision over social media posts allows rumours and fake news to propagate rapidly in society.

The proliferation of rumours and fake news has become a serious concern due to the rapid growth of internet technology and online media platforms. These issues often trigger anxiety or scepticism and entice readers to share them without verification [3, 49]. Daily, various rumours, and fake news on different topics spread through online media and significantly influence people's opinions. Therefore, there is an urgent need for automated detection of rumours or fake news, as manual fact-checking is challenging and time-consuming.

2.2 Traditional machine-learning-based approaches

Some previous studies utilised traditional machine-learning methods to identify rumours automatically. Traditional machine-learning algorithms require fewer computational resources, making them ideal for limited training time scenarios. They can deliver good results with smaller labelled datasets, which is helpful when data is limited or costly to obtain. Additionally, these methods enable domain experts to apply their knowledge and perform feature engineering to extract relevant text features, which is especially useful for tasks like rumour classification that benefit from domain-specific insights. [1–5].

However, traditional machine learning has its limitations. It lacks comprehensive context understanding as it treats words in isolation and fails to capture the contextual relationships between words within sentences or documents. Consequently, its ability to grasp the entire context of the text, especially in rumour detection tasks involving complex language structures, is limited. Additionally, traditional methods heavily rely on handcrafted features, which may overlook the rich, complex semantics in text data. Moreover, manually extracting features can be time-consuming, and it becomes more challenging when certain features are missing due to a user's security settings. Such conditions can impact the effectiveness of the feature-based approach.

2.3 Deep learning-based approaches

Several studies have proposed a deep learning framework with neural networks to address the limitations of traditional machine learning. This framework allows direct learning of hierarchical representations from raw text, eliminating the need for extensive feature engineering and enabling the automatic extraction of crucial features. Deep learning has also demonstrated promising outcomes in various text classification tasks. In the context of false information detection, the widely implemented neural network framework approaches on detecting rumour are recurrent neural network (RNN) [6–8], long short-term memory (LSTM) [14–16], and convolutional neural network (CNN) [9–13].

The following recent studies utilised neural network-based hybrid approaches and achieved new state-of-the-art results, making them benchmark methods for comparing the performance of the proposed methods. Wu and Rao worked on rumour detection by utilising interaction between the features of rumours. The study proposed models called "Adaptive Interaction Fusion Networks (AIFN)" and "Gated Adaptive Interaction Networks (GAIN)" to identify true and false rumours on Twitter [50]. These techniques were used to identify interaction features among the tweets and capture semantic conflict in posts and comments. The study evaluated their models on the Pheme dataset [15]. Later a "Decision Tree-based CoAttention model (DTCA)" was proposed to extend this study by utilising the interaction of credible comments as evidence to detect the truth of the tweet [51]. The study reported 82.46% accuracy and 82.5% F1 score.

Wang et al. [25] aimed to detect rumour by considering a background hidden knowledge in the post's text content and proposed a "Knowledge-driven Multimodal Graph Convolutional Network". This approach combined textual, conceptual, and visual data to represent the semantic information into a unified framework for fake news identification. It used Pheme dataset [3] to validate the model. The study reported above 87% for all evaluation parameters including, accuracy, precision, recall, and F1 score.

Lu and Li [24] aimed to predict whether the source of tweets was fake by using user profiles and social interactions. A graph-aware coattention network (GCAN) was proposed based on neural network models to depict users interaction and capture the relationship between the source tweet, its propagation, and user interaction to generate a prediction. The model was evaluated using Twitter 15 and Twitter 16 datasets [6]. Accuracy of 87% and 90% on Twitter 15 and Twitter 16 datasets was reported, respectively.

Wang and Guo encoded tweets' sentiment information and word representation with a two-layer cascaded gated recurrent unit (CGRU) to detect rumours [52]. The model was validated using Twitter 16 datasets [6]. The model achieved 88.5% accuracy which outperformed earlier studies. Another study [11] classified tweets as rumour or non-rumour using a dual convolutional neural network (DCNN) technique using the inherent features of the information set. Their study used Twitter 15 and Twitter 16 datasets [6] to evaluate the model and reported F1 scores of 86% and 87%, respectively.

Zhanh et al. proposed a lightweight propagation path aggregating (PPA) neural network for rumour embedding and classification [19]. They first modelled the propagation structure of each rumour as an independent set of propagation paths in which each path represents the source post in a different conversation context. Then, they aggregated all paths to obtain the representation of the whole propagation structure. Twitter 15, Twitter 16, and Pheme datasets were used to evaluate their model and achieved accuracies of 87.3%, 88.7%, and 80.3% for Twitter 15, Twitter 16, and Pheme, respectively.

2.4 BERT-based language models

The emergence of BERT significantly improved the solutions related to natural language processing (NLP). BERT is a pre-trained language model trained on vast unlabelled data from the BooksCorpus that includes 800 million words and English Wikipedia with 2500 million words without any genuine training objective. Therefore, it can be fine-tuned for many NLP tasks. BERT can understand a word's contextual meaning by considering words from both the left and the right sides. Furthermore, it can represent words and sentences that are converted into numeric vectors [34]. There are two standard architectures of BERT: $\text{BERT}_{\text{BASE}}$ and $\text{BERT}_{\text{LARGE}}$. In general, $\text{BERT}_{\text{BASE}}$ has 12 encoder layers with 110 million parameters and 768 hidden layers, and $\text{BERT}_{\text{LARGE}}$ has 16 encoder layers with 340 million parameters and 1024 hidden layers.

Despite the benefit and ability of the pre-trained language model, there are issues regarding training, memory consumption, and computational power due to a large amount of training data. Therefore, some researchers introduced different LMs to address these issues, such as RoBERTa [40] to address a training model issue and DistilBERT [43] to deal with the memory and computational inefficiency problems.

Liu et al. examined the effect of hyperparameter tuning and training size on BERT architecture [40]. They revealed that BERT was significantly under-trained and introduced an improved BERT training model called RoBERTa (robustly optimised BERT approach). The modified BERT's training procedures were, including;

(1) Training the model with a larger dataset;
(2) Removing the next sentence prediction objective;
(3) Training in a more extended sequence; and
(4) Changing the masking pattern dynamically to the training data. RoBERTa was trained by following $\text{BERT}_{\text{LARGE}}$ architecture. However, unlike BERT, which initially trained using 16GB of the dataset, a more extensive training dataset containing 160GB of text was used to train RoBERTa. In addition, RoBERTa was trained for more iterations of 300,000, which was later extended to 500,000. RoBERTa consistently outperformed BERT in all individual tasks on the standard GLUE benchmark [53].

Although BERT and RoBERTa are leading in the current NLP research, the large models raise models pose challenges with computational costs and requirements. To address this, Sanh et al. proposed a lighter and faster LM based on BERT architecture, using knowledge distillation to compress the model and reduce these computational issues [43]. Furthermore, they trained the compressed model to reproduce the behaviour of the large model. The compressed model is called DistilBERT. It has 66 million parameters, 40% fewer than BERT, and trains 60% faster. Experiment results showed that, despite its smaller size, DistilBERT performs well on several NLP tasks and is lightweight enough to run on mobile devices.

Researchers in information credibility have found that using BERT-based language models for classifying misinformation yields excellent performance compared to standard machine-learning methods [54–56]. Furthermore, other researchers compared LMs' performance to evaluate the cross-source failure problem in the current misinformation detection methods. They focussed on finding generalisable representation to make the classification model more applicable to real-world data [57]. They examined the cross-source generalisability by choosing one dataset for training set and others for testing sets.

Pelrine et al. evaluated the performance of a few pre-trained language models on detecting rumours [58]. However, to establish a solid benchmark, they used different data distributions in each dataset, following previous studies to ensure a fair comparison. They fine-tuned the language models and used a fully connected layer as the classifier. The results showed that this approach outperformed state-of-the-art models. Therefore, Pelrine's et al. study is included as a current state-of-the-art model and compared with the proposed methods.

While prior research has addressed rumour detection using text embeddings, our study fills existing gaps by offering a new perspective. We build on established theories in both areas and adapt them to address the unique challenges of Twitter data. Our framework supports our experiments and advances these fields.

3 Material and method

Our methodological approach contributes to the theoretical foundations of rumour detection and text embeddings. We carefully designed our experiments to answer our research questions while introducing innovative techniques to address the challenges. This section describes the proposed models, the datasets used for validation, and the experimental steps taken. The study focuses on comparing and evaluating text embeddings from three BERT-based models-RoBERTa, BERT$_{\mathrm{BASE}}$, and DistilBERT-in distinguishing rumour from non-rumour tweets and classifying true and false rumours using a neural network classifier. Various datasets were then used to validate the performance of the proposed methods.

3.1 The proposed model

This study utilises the fine-tuned language model as an encoder to represent text into vectors and apply neural network models as a classifier. The architecture of the proposed method is presented in Fig. 1.

The fine-tuned language model encoded sentences into vectors, followed by classification using two neural network techniques: multilayer perceptron (MLP) and residual network on convolutional neural network (ResNet-CNN). MLP is a neural network that comprises more than one perceptron. Generally, it consists of an input layer to receive the signal, an output for predicting the input, and an arbitrary number of hidden layers between the input and output [59]. This study uses two and four layers of MLP and performs regularisation and dropout on the model.

On the other hand, ResNet-CNN serves as an enhanced model to tackle the vanishing gradient issue that often arises in convolutional neural networks (CNNs) with deep layers. A CNN resembles an MLP but incorporates various layers in its hidden layer, encompassing convolutional, pooling, fully connected, and regularisation layers. These additional layers significantly contribute to the success of most deep neural networks. However, a study by He et al. (2016) empirically established a threshold for the maximum depth of these additional layers in the CNN model. To address this, a novel neural layer known as the residual network (ResNet) was introduced to mitigate the challenge of deeper networks. ResNet consists of residual blocks with skip connections that link outputs from earlier layers to those of later stacked layers. These skip connections effectively address the

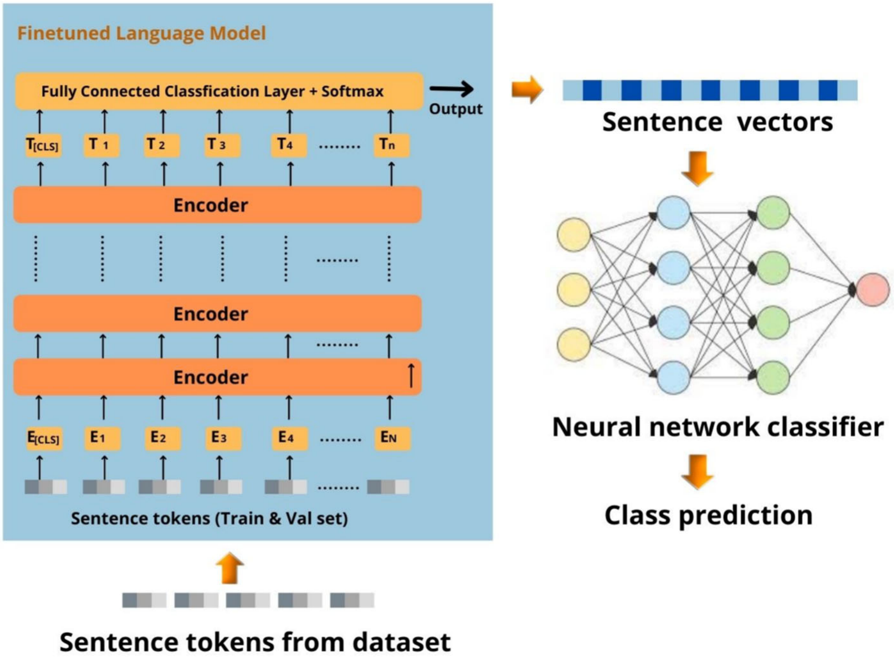

Finetuned Language Model
Fully Connected Classfication Layer + Softmax
Output
T[CLS]
T1
T2
T3
T4
....
Tn
Encoder
Encoder
Encoder
E[CLS]
E1
E2
E3
E4
EN
Sentence tokens (Train & Val set)
Sentence vectors
Neural network classifier
Class prediction
Sentence tokens from dataset

Fig. 1 The proposed architecture for detecting information credibility uses finetuned language models to encode sentences into vectors, with various neural network techniques as classifiers

vanishing gradient issue in CNNs and enhance the performance of neural networks with increased layer depth [60]. In this study, we employed 10-layer ResNet-CNN for classification purposes.

3.2 Datasets

Short-text (tweets) datasets were used to validate the proposed method's performance. Regarding the splitting data technique, this study suggests taking the same distribution of data for all datasets to make a standard testing procedure. It is different from many existing studies that took a different distribution based on the state-of-the-art results of each dataset. The datasets are explained in detail in the following subsections.

3.2.1 Pheme datasets

Pheme contains 5791 tweets about five breaking news topics where 1969 tweets are labelled as rumours and 3822 are labelled as non-rumours by journalists [3] and extended by adding with four additional news topics, meaning that Pheme contained 6425 tweets about nine news events in total. The journalists verified whether these tweets were rumour (2402 tweets) or non-rumour (4023 tweets). Then, rumour tweets are further classified and labelled as true-rumour (1067 tweets), false-rumour (638 tweets), or unverified-rumour (697 tweets) [15]. However, this study does not consider the data with unverified-rumour labels as it is beyond the scope of this study.

Pheme datasets are used in a different manner in different state-of-the-art techniques. Therefore, we evaluated the proposed models, in the same manner, to make a fair comparison with existing studies. These detailed subdatasets are explained below:

- Pheme R/NR: It consists of 6425 tweets from Pheme dataset where 2402 tweets are categorised as rumour tweets while 4023 are categorised as non-rumour tweets.
- Pheme T/F: It contains only 1705 tweets that are marked as true or false rumours and taken from Pheme dataset. It consists of 1067 and 638 tweets labelled as true rumours and false rumours, respectively.

3.2.2 Twitter 15 and Twitter 16

Twitters 15 and 16 are publicly available datasets containing 1490 and 818 tweets. The journalists labelled each tweet as rumour or non-rumour. Afterwards, each rumour tweet is classified and labelled as a true rumour, false rumour, or unverified rumour [6]. Like the Pheme dataset, this study did not consider the data with unverified-rumour labels. This study uses four versions of this dataset, as follows:

- Twitter 15 R/NR: It contains 1490 tweets from the Twitter 15 dataset where 1118 are labelled as a rumour while 372 are labelled as non-rumour.
- Twitter 16 R/NR: It consists of 818 tweets from the Twitter 16 dataset marked as a rumour (613 tweets) or non-rumour (205 tweets).
- Twitter 15 T/F: It contains tweets labelled as true rumour (375 tweets) or false rumour (370 tweets) from the Twitter 15 dataset.
- Twitter 16 T/F: It comprises 410 tweets from Twitter 16 dataset, which are marked as true rumour (205 tweets) or false rumour (205 tweets).

3.2.3 Combined dataset

To generalise the performance of the proposed method, we combined the dataset as suggested by [58]. This study generalised combined short-text datasets with the same label from Pheme, Twitter 15, and Twitter 16 datasets to obtain a generalised performance result in classifying rumour and non-rumour tweets and distinguishing true- and false-rumour tweets.

A single combined dataset with rumour and non-rumour labels named Combined R/NR was created from Pheme R/NR, Twitter 15 R/NR, and Twitter 16 R/NR to validate the model's performance in classifying rumour and non-rumour tweets. Hence, the Combined R/NR dataset comprises 4600 non-rumour tweets and 4133 rumour tweets. Similarly, Pheme T/F, Twitter 15 T/F, and Twitter 16 T/F datasets were incorporated into a single dataset named Combined T/F dataset. The dataset consists of 1213 false-rumour tweets and 1646 true-rumour tweets. Table 1 describes the more detailed distribution of the datasets used in this study.

3.3 Experimental procedure

The experiments used an RTX 2080 GPU to train each model. The experimental setup for identifying rumours/non-rumours tweets, and true/false rumours using BERT-LMs and neural network models is shown in Fig. 2.

As a first step, a dataset was split into training, validation, and testing sets. From each dataset, 10% of the data is reserved first for testing. The remaining data was split into 75% and 25% for training and validation sets, respectively. For the combined datasets, 10% of the data was taken from each dataset for the testing set before blending them into a single Combined T/F dataset or Combined R/NR dataset for training and validation. The rest of the combined dataset was split into 3:1 proportions for the training and validation sets, respectively.

Table 1 Distribution of labels in the datasets used in this study

Datasets and labels	Pheme [3, 15]	Twitter 15 [6]	Twitter 16 [6]	Combined R/NR	Combined T/F
Non-rumours	4023	372	205	4600	–
Rumour	2402	1118	613	4133	–
False rumours	638	370	205	–	1213
True rumours	1067	374	205	–	1646
Unverified (not used)	*697*	*374*	*203*	–	–

Bold represent best results, underlined and italicized represent the second-best and third-best results in tables

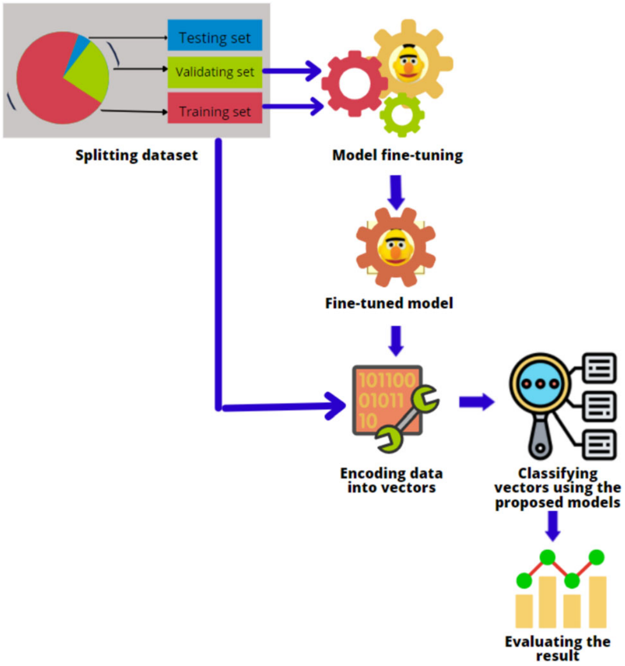


Fig. 2 The experiment setup to identify rumours/non-rumour tweets and true/fake news using the proposed model

Table 2 State-of-the-art results for rumour and fake news detection for each dataset

Datasets	Sources	Splitting data strategy	Best performance			
			A (%)	P (%)	R (%)	F1 (%)
Pheme R/NR	[63]	Not provided	83.36	83.59	99.64	90.91
Pheme T/F	[51]	70:10:20 for train:val:test	82.46	79.08	86.24	82.5
	[58]	70:10:20 for train:val:test	–	–	–	93.2 ± 0.9
Twitter 15 T/F	[24]	70:30 for train:test	87.67	82.57	82.95	82.50
	[58]	70:10:20 for train:val:test	–	–	–	94.4 ± 0.8
Twitter 16 T/F	[24]	70:30 for train:test	90.84	75.94	76.32	75.93
	[58]	70:10:20 for train:val:test	–	–	–	95.7 ± 2.8
Twitter 15 R/NR	[64]	Not provided	86.3	–	–	–
	[11]	60:40 for train:test	86	–	–	–
Twitter 16 R/NR	[52]	75:25 for train:test	88.5	–	–	–
	[11]	60:40 for train:test	87	–	–	–
Covid T/F	[65]	Not provided	75.43	66.22	56.33	60.9

A accuracy, *P* precision, *R* recall, *F1* F1 score

The next phase involved fine-tuning the BERT-based language model (LM) using the Huggingface library [61], with the labelled data as input. We employed the Adam optimiser, with a learning rate set at 5e-5 and a batch size of eight. Refer to [62] that reported the optimum iteration of fine-tuning BERT to save computational cost is 20 epochs; hence, our experiments were conducted for a total of 20 epochs. In addition, we proceeded to refine-tune the BERT language models by increasing the number of epochs to 40. This was undertaken to assess whether this extended fine-tuning period could enhance the performance of our models. We set the mean pooling layer as a pooler to encode tweets into vectors. Primarily, these vectors were used to train the classifier models.

In the classification phase, we configured the learning rate at 2e–4, the batch size at 512, and chose a maximum of 1000 epochs. The models were trained using the Adam optimiser, with binary cross-entropy employed as the

Table 3 Comparison of the proposed method with state-of-the-art techniques for Pheme R/NR dataset

Classifier models			Best results on Pheme R/NR (6425 tweets; 2402 R, 4023 NR)				Confusion matrix			
SOTA models			A (%)	P (%)	R (%)	F1 (%)	TP	FP	FN	TN
[66]	Split dataset is not provided		83.36	83.59	99.64	90.91				
Proposed Models [10%: testing, 75% and 25% of rest of the data for training and validation, respectively]										
2MLP	BERT	Train	100	100	100	100	3185	0	0	1151
		Val	88.24	85.09	84.5	84.79	993	91	81	297
		Test	89.16	86.68	84.92	85.79	433	40	28	126
	RoBERTa	Train	96.2	93.92	96.99	95.43	3035	15	150	1136
		Val	88.24	84.29	87.38	85.81	958	56	116	332
		Test	*89.16*	*85.54*	*87.61*	*87.61*	419	26	42	140
	DistilBERT	Train	99.61	99.27	99.73	99.50	3168	0	17	1151
		Val	88.51	85.23	85.35	85.29	989	83	85	305
		Test	86.76	82.91	83.29	83.10	418	40	43	126
4MLP	BERT	Train	100	100	100	100	3185	0	0	1151
		Val	88.24	85.66	83.43	84.53	1006	104	68	284
		Test	89.63	88.11	84.47	86.25	433	40	28	126
	RoBERTa	Train	99.68	99.43	99.75	99.59	3172	1	13	1150
		Val	88.30	84.73	85.78	85.25	979	76	95	312
		Test	**89.95**	**86.88**	**87.58**	**87.23**	427	29	34	137
	DistilBERT	Train	99.12	98.45	99.35	98.90	3149	2	36	1149
		Val	88.30	85.05	84.88	84.97	990	87	84	301
		Test	86.92	83.34	82.82	83.08	422	43	39	123
10L-ResNet-CNN	BERT	Train	100	100	100	100	3185	0	0	1151
		Val	87.76	84.05	85.08	84.56	975	80	99	308
		Test	88.52	85.18	85.45	85.31	424	35	37	131
	RoBERTa	Train	99.68	99.43	99.75	99.59	3172	1	13	1150
		Val	87.89	84.05	85.92	84.98	968	71	106	317
		Test	88.36	84.69	86.11	85.39	419	31	42	135
	DistilBERT	Train	99.77	99.65	99.76	99.71	3178	3	7	1148
		Val	87.55	84.13	83.79	83.96	986	94	88	294
		Test	87.40	84.17	82.95	83.56	426	44	35	122

A Accuracy, *P* precision, *R* recall, *F1*:F1 score, *TP* true positive, *FP* false positive, *FN* false negative, *TN* true negative, *SOTA* State-of-the-art

Bold value represent best results, where as underlined and italicized represent the second-best and third-best results in tables

loss function. To assess the models' performance comprehensively, we conducted evaluations by calculating key metrics including accuracy, precision, recall, and F1 score.

4 Results and discussion

This section outlines the experimental outcomes of the proposed models and compares their evaluation metrics to those of the SOTA models for each dataset. We have tabulated the SOTA models and their respective performances in Table 2. As various SOTA models have reported different metrics in their studies, our research embraces a comprehensive approach by presenting all evaluation metrics, encompassing accuracy, precision, recall, F1 score, and confusion matrix. This detailed evaluation offers a complete understanding of the models' performance and highlights our focus on both false positives and false negatives in the assessment. Precision, recall, and accuracy are all vital evaluation metrics, and their significance depends on the specific objectives and

Table 4 Comparison of the proposed method with state-of-the-art techniques for Twitter 15 R/NR dataset

Classifier models			Best results on Twitter 15 R/NR (1490 tweets; 1118 R, 372 NR)				Confusion matrix			
SOTA models			A (%)	P (%)	R (%)	F1 (%)	TP	FP	FN	TN
[64]	Split dataset is not provided		86.3	–	–	–				
[11]	60:40 for train:test		86	–	–	–				
Proposed Models [10%: testing, 75% and 25% of rest of the data for training and validation, respectively]										
2MLP	BERT	Train	100	100	100	100	253	0	0	771
		Val	84.02	83.27	73.54	78.10	46	10	44	238
		Test	88.28	88.98	78.00	83.13	18	2	13	95
	RoBERTa	Train	100	100	100	100	253	0	0	771
		Val	87.87	86.31	81.47	83.82	61	12	29	236
		Test	89.06	86.29	82.91	84.56	22	5	9	92
	DistilBERT	Train	100	100	100	100	253	0	0	771
		Val	84.62	81.86	76.77	79.23	54	16	36	232
		Test	86.72	82.50	80.26	81.37	21	7	10	90
4MLP	BERT	Train	100	100	100	100	253	0	0	771
		Val	84.02	83.27	73.54	78.10	46	10	44	238
		Test	88.28	88.98	78.00	83.13	18	2	13	95
	RoBERTa	Train	100	100	100	100	253	0	0	771
		Val	87.87	86.31	81.47	83.82	61	12	29	236
		Test	**89.06**	**86.29**	**82.91**	**84.56**	22	5	9	92
	DistilBERT	Train	100	100	100	100	253	0	0	771
		Val	84.91	82.18	77.33	79.68	55	16	35	232
		Test	86.72	82.50	80.26	81.37	21	7	10	90
10L-ResNet-CNN	BERT	Train	100	100	100	100	253	0	0	771
		Val	83.73	81.76	74.05	77.71	48	13	42	235
		Test	86.72	84.78	76.97	80.69	18	4	13	93
	RoBERTa	Train	100	100	100	100	253	0	0	771
		Val	87.57	86.03	80.91	83.39	60	12	30	236
		Test	*89.06*	*86.29*	*82.91*	*84.56*	22	5	9	92
	DistilBERT	Train	99.90	99.80	99.94	99.87	253	1	0	770
		Val	85.50	81.86	80.21	81.03	62	21	28	227
		Test	85.94	80.52	83.04	81.76	24	11	7	86

A Accuracy, *P* precision, *R* recall, *F1* F1 score, *TP* true positive, *FP* false positive, *FN* false negative, *TN* true negative, *SOTA* state-of-the-art

Bold value represent best results, where as underlined and italicized represent the second-best and third-best results in tables

the implications of false positives and false negatives in real-world applications. If the primary concern is to minimise the dissemination of false information, it is advisable to prioritise precision. Moreover, recall gains importance if the goal is to capture as many actual rumours as possible. While the F1 score offers a balanced metric that considers both precision and recall, it proves useful to have a balance between minimising false positives and ensuring extensive coverage of actual rumours. Furthermore, accuracy offers a comprehensive measure of the model's correctness in classifying both rumours and non-rumours, which holds importance as a general indicator of overall performance across both classes, especially when the class distribution is relatively balanced.

The comparative performance of all the models in the rumour and non-rumour datasets can be found in Tables 3, 4, 5, 6 and 7, while Tables 8, 9 and 10 depict the results for the true- and false-rumour datasets. In each table, the first and second rows provide details about the state-of-the-art models and their data-splitting strategies

Table 5 Comparison of the proposed method with state-of-the-art techniques for Twitter 16 R/NR dataset

Classifier models			Best results on Twitter 16 R/NR (818 tweets; 613 R, 205 NR)				Confusion matrix			
SOTA models			A (%)	P (%)	R (%)	F1 (%)	TP	FP	FN	TN
[64]	Split dataset is not provided		88.5	–	–	–				
[11]	60:40 for train:test		87	–	–	–				
Proposed Models [10%: testing, 75% and 25% of rest of the data for training and validation, respectively]										
2MLP	BERT	Train	100	100	100	100	140	0	0	405
		Val	87.5	86.035	77.188	81.37	26	5	19	142
		Test	80.25	77.91	63.36	69.88	6	2	14	59
	RoBERTa	Train	100	100	100	100	140	0	0	405
		Val	86.46	81.35	80.36	80.85	31	11	14	136
		Test	**93.83**	**96.21**	**87.50**	**91.65**	14	0	6	61
	DistilBERT	Train	100	100	100	100	140	0	0	405
		Val	88.02	84.60	80.61	89.06	30	8	15	139
		Test	82.72	78.25	71.72	74.84	10	4	10	57
4MLP	BERT	Train	100	100	100	100	140	0	0	405
		Val	87.5	86.03	77.18	81.37	26	5	19	142
		Test	80.25	77.91	63.36	69.89	6	2	14	59
	RoBERTa	Train	100	100	100	100	140	0	0	405
		Val	86.98	82.24	80.70	81.46	31	11	14	136
		Test	92.59	95.52	85	89.95	14	0	6	61
	DistilBERT	Train	100	100	100	100	140	0	0	405
		Val	88.02	85.78	79.07	82.29	28	6	17	141
		Test	83.95	81.11	72.54	76.58	10	3	10	58

A Accuracy, *P* precision, *R* recall, *F1* F1 score, *TP* true positive, *FP* false positive, *FN* false negative, *TN* true negative, *SOTA* state-of-the-art

Bold value represent best results, where as underlined and italicized represent the second-best and third-best results in tables

that achieved top-notch performance. The best result from each column is highlighted in bold, whereas the second-best and third-best results are indicated with underlined and italicised font, respectively.

4.1 Comparison of results with the SOTA on rumours/non-rumours datasets

Tables 3, 4 and 5, and Figs. 3, 4 and 5 present the experimental results of the proposed methods for classifying rumours and non-rumours on Pheme, Twitter 15, and Twitter 16 datasets, respectively. These tables demonstrate a consistent finding that generally classifier models using text embedding from RoBERTa exhibit better performance than other language models. It is understandable as RoBERTa is the most complex and largest language model and has a higher number of parameters compared to BERT and DistillBERT.

Additionally, RoBERTa's superior performance may be due to several factors: extensive training on a larger, more diverse dataset, careful hyperparameter tuning, dynamic masking during pre-training, and the exclusion of the next sentence prediction (NSP) task. These factors enable RoBERTa to capture rich contextual information, complex language patterns, and long-range dependencies in text. Therefore, RoBERTa generates superior text representations (embeddings) that are beneficial in the classification of tweets compared to BERT and DistillBERT.

Analysing the performance of the proposed models on the Pheme R/NR dataset, we observed that generally, all the proposed models outperformed the current state-of-the-art model, except 10 L-ResNet-CNN coupled with DistillBERT embedding, as depicted in Table 3 and Fig. 3. Although the model does not surpass the SOTA, the 10 L-ResNet-CNN with DistillBERT embeddings achieved results close to the existing benchmark. Additionally,

Table 6 Comparison of the proposed method with state-of-the-art techniques for combined R/NR dataset

Classifier models			Best results on Combined R/NR (8733 tweets; 4133 R, 4600 NR)				Confusion matrix			
			A (%)	P (%)	R (%)	F1 (%)	TP	FP	FN	TN
Proposed Models [10%: testing, 75% and 25% of rest of the data for training and validation, respectively]										
2MLP	BERT	Train	100	100	100	100	2546	0	0	3359
		Val	96.74	96.55	96.82	96.69	830	43	22	1097
		Test	96.89	96.46	97.17	96.81	325	21	5	485
	RoBERTa	Train	98.27	98.35	98.13	98.24	2472	28	74	3331
		Val	96.84	96.89	96.64	96.77	812	23	40	1117
		Test	*97.37*	*97.30*	*97.19*	*97.24*	318	10	12	496
	DistilBERT	Train	99.95	99.94	99.96	99.95	2546	3	0	3356
		Val	96.79	96.66	96.79	96.73	825	37	27	1103
		Test	96.29	95.85	96.52	96.18	322	23	8	483
4MLP	BERT	Train	99.98	99.99	99.98	99.98	2545	0	1	3359
		Val	96.79	96.64	96.82	96.73	827	39	25	1101
		Test	97.25	96.90	97.41	97.16	324	17	6	489
	RoBERTa	Train	98.66	98.66	98.61	98.64	2501	34	45	3325
		Val	97.14	97.06	97.10	97.08	825	30	27	1110
		Test	97.49	97.26	97.50	97.38	322	13	8	493
	DistilBERT	Train	99.20	99.19	99.19	99.19	2523	24	23	3335
		Val	96.74	96.66	96.68	96.67	820	33	32	1107
		Test	96.41	96.09	96.46	96.27	319	19	11	487
10L-ResNet-CNN	BERT	Train	99.78	99.78	99.77	99.78	2538	5	8	3354
		Val	96.54	96.41	96.53	96.47	822	39	30	1101
		Test	96.77	96.41	96.91	96.66	322	19	8	487
	RoBERTa	Train	99.71	99.74	99.67	99.71	2530	1	16	3358
		Val	96.99	96.95	96.89	96.92	820	28	32	1112
		Test	**97.61**	**97.37**	**97.66**	**97.51**	323	13	7	493
	DistilBERT	Train	99.17	99.08	99.24	99.16	2540	43	6	3316
		Val	96.79	96.59	96.91	96.75	833	45	19	1095
		Test	96.53	96.13	96.71	96.42	322	21	8	485

A Accuracy, *P* precision, *R* recall, *F1* F1 score, *TP* true positive, *FP* false positive, *FN* false negative, *TN* true negative, *SOTA* state-of-the-art

Bold value represent best results, where as underlined and italicized represent the second-best and third-best results in tables

the 2MLP and 4MLP models performed slightly better than ResNet-CNN with BERT-based embeddings, with variations of only 1% to 3%.

Shifting our focus to the Twitter 15 R/NR dataset, as presented in Table 4 and Fig. 4, the findings indicate that on Twitter 15 R/NR dataset, all the proposed models consistently demonstrated exceptional performance, surpassing the current state-of-the-art models [11, 25, 64]. The only exception to this trend is again, 10 L-ResNet-CNN model coupled with DistillBERT embedding, which achieved comparable results with the current benchmark. On the Twitter 16 R/NR dataset, as highlighted in Table 5, it is evident that all classifier models employing RoBERTa as the embedder delivered the highest levels of performance, consistently outperforming classifiers coupled with BERT and DistilBERT as embedders by a significant margin, with differences reaching up to 13%.

While all our proposed models to classify rumours and non-rumours achieved high performance, often surpassing or achieving results comparable to the current SOTA model, we observed oscillations and dips line on accuracy curves of MLP models in Fig. 3 and performance disparities between the training and validation-set

Table 7 Cross-validation evaluation of the proposed models on rumour/non-rumour dataset

Dataset	Classifier models			Fivefold Cross-validation on R/NR datasets			
SOTA models				A (%)	P (%)	R (%)	F1 (%)
[66]	Split dataset is not provided			83.36	83.59	99.64	90.91
Pheme R/NR	4MLP	BERT	Fold-1	86.12	81.77	84.97	83.34
			Fold-2	88.04	84.60	84.73	84.67
			Fold-3	88.20	84.77	85.04	84.90
			Fold-4	88.84	85.31	86.63	85.96
			Fold-5	88.68	86.75	83.24	84.96
			Avg	87.97	84.64	84.92	84.77
	4MLP	RoBERTa	Fold-1	88.68	85.50	85.36	85.43
			Fold-2	88.52	85.10	85.64	85.37
			Fold-3	87.72	83.78	85.87	84.81
			Fold-4	89.31	85.87	87.34	86.60
			Fold-5	88.52	84.77	86.79	85.77
			Avg	88.55	85.00	86.20	85.60
Twitter R/NR	4MLP	BERT	Fold-1	88.28	88.98	78.00	83.13
			Fold-2	88.28	88.98	78.00	83.13
			Fold-3	88.28	88.98	78.00	83.13
			Fold-4	85.16	80.21	78.13	79.16
			Fold-5	88.28	88.98	78.00	83.13
			Avg	87.66	87.23	78.03	82.34
	4MLP	RoBERTa	Fold-1	89.06	86.29	82.91	84.56
			Fold-2	89.06	86.29	82.91	84.56
			Fold-3	89.06	86.29	82.91	84.56
			Fold-4	89.06	86.29	82.91	84.56
			Fold-5	89.06	86.29	82.91	84.56
			Avg	89.06	86.29	82.91	84.56
	10L-ResNet-CNN	RoBERTa	Fold-1	88.28	83.85	84.59	84.22
			Fold-2	89.06	85.62	84.00	84.80
			Fold-3	89.06	86.29	82.91	84.56
			Fold-4	88.28	84.25	83.49	83.87
			Fold-5	87.50	83.39	81.88	82.62
			Avg	88.44	84.68	83.37	84.02

A Accuracy, *P* precision, *R* recall, *F1* F1 score, *SOTA* state-of-the-art

results in some models, as shown in Tables 3, 4 and 5. This may indicate overfitting issues and may need further evaluation.

It can be observed from Figs. 3, 4 and 5 that the appearance of oscillations is more in the accuracy curve of the classifiers model when training on the Pheme dataset which has more imbalanced classes compared to Twitter 15 and Twitter 16 (see Table 1). This gives rise to the suspicion that the oscillation issues may be attributed to the relatively small and unbalanced labels of the rumour/non-rumour datasets. In a small-size dataset, the model may encounter a limited variety of examples, increasing the likelihood of memorising noise present in the training set. As a result, the model may struggle to generalise effectively to new and unseen data, leading to overfitting. Another reason is the complexity of the MLP model compared to the given dataset, so the model may try to fit the training data too closely, capturing noise rather than the underlying patterns. Suboptimal selection of hyperparameters, like learning rate, batch size, or number of epochs, may also play a role. This challenge is further complicated by the black-box nature of deep learning networks.

Table 8 Comparison of the proposed method with state-of-the-art techniques for Pheme T/F dataset

Classifier models			Best results on Pheme T/F (1705 tweets; 1067 T, 638 F)				Confusion matrix			
SOTA models			A (%)	P (%)	R (%)	F1 (%)	TP	FP	FN	TN
DTCA [51]	70:10:20 for train:val:test		82.46	79.08	86.24	82.5	–	–	–	–
Pelrine et al.[58]	70:10:20 for train:val:test		–	–	–	93.2 ±0.9	–	–	–	–
Proposed Models [10%: testing, 75% and 25% of rest of the data for training and validation, respectively]										
2MLP	BERT	Train	100	100	100	100	434	0	0	742
		Val	85.98	85.50	84.69	85.09	113	22	30	206
		Test	87.98	87.87	86.56	87.21	49	7	12	90
	RoBERTa	Train	100	100	100	100	434	0	0	742
		Val	91.64	91.68	90.60	91.13	123	11	20	217
		Test	91.14	90.65	90.65	90.65	54	7	7	90
	DistilBERT	Train	91.14	90.65	90.65	90.65	54	7	7	90
		Val	85.98	85.41	84.82	85.11	114	23	29	205
		Test	86.08	85.47	85.01	85.24	49	10	12	87
4MLP	BERT	Train	99.83	99.87	99.77	99.82	432	0	2	742
		Val	85.71	86.01	83.55	84.77	106	16	37	212
		Test	88.61	89.54	86.46	87.97	47	4	14	93
	RoBERTa	Train	100	100	100	100	434	0	0	742
		Val	92.18	92.26	91.16	91.71	124	10	19	218
		Test	**91.14**	**90.65**	**90.65**	**90.65**	54	7	7	90
	DistilBERT	Train	96.60	97.11	95.63	96.37	399	5	35	737
		Val	86.25	86.80	83.99	85.37	106	14	37	214
		Test	86.71	87.59	84.31	85.92	45	5	16	92
10L-ResNet-CNN	BERT	Train	98.30	98.05	98.32	98.18	427	13	7	729
		Val	85.18	84.38	84.29	84.33	115	27	28	201
		Test	84.18	83.52	82.85	83.18	47	11	14	86
	RoBERTa	Train	100	100	100	100	434	0	0	742
		Val	89.76	89.98	88.28	89.12	117	12	26	216
		Test	*90.51*	*90.09*	*89.83*	*89.96*	53	7	8	90
	DistilBERT	Train	97.96	98.37	97.28	97.83	411	1	23	741
		Val	86.52	86.55	84.73	85.63	110	17	33	211
		Test	84.81	85.19	82.46	83.80	44	7	17	90

A Accuracy, *P* precision, *R* recall, *F1* F1 score, *TP* true positive, *FP* false positive, *FN* false negative, *TN* true negative, *SOTA* state-of-the-art
Bold value represent best results, where as underlined and italicized represent the second-best and third-best results in tables

To address this challenge, first, we augmented the dataset by combining the Pheme R/NR, Twitter 15 R/NR, and Twitter 16 R/NR datasets into a single dataset referred to as “Combined R/NR”, which then considered has more balanced labels. Subsequently, we trained all the proposed models on this augmented dataset, and the results are presented in Table 6. As observed in Table 6, all the proposed models exhibited exceptional performance, particularly when coupled with BERT models as the embedders. Furthermore, Table 6 demonstrates that while RoBERTa achieved the best results, the performance disparity compared to BERT and DistilBERT was minimal, typically within a range of approximately 1%.

We also performed fivefold cross-validation to evaluate the model’s generalisation capability on the best classifier models on the Pheme R/NR dataset and present the result in Table 7. Table 7 demonstrates that our models performed stable results on each fold and obtained a higher accuracy than the SOTA model. Up to this point, we note that RoBERTa, as the most complex and largest language model, may represent text better than BERT and DistilBERT, though the difference is minor. Therefore, all the proposed classifier models generally

Table 9 Comparison of the proposed method with state-of-the-art techniques for Twitter 15 T/F dataset

Classifier models			Best results on Twitter 15 T/F (754 tweets; 374 T, 320 F)				Confusion matrix			
SOTA models			A (%)	P (%)	R (%)	F1 (%)	TP	FP	FN	TN
Gcan [24]	70:10:20 for train:val:test		87.67	82.57	82.95	82.50	–	–	–	–
Pelrine et al.[58]	70:10:20 for train:val:test		–	–	–	94.4 ±0.8	–	–	–	–
Proposed Models [10%: testing, 75% and 25% of rest of the data for training and validation, respectively]										
2MLP	BERT	Train	100	100	100	100	251	0	0	246
		Val	89.47	89.57	89.65	89.61	67	11	5	69
		Test	91.40	91.40	91.40	91.40	43	4	4	42
	RoBERTa	Train	100	100	100	100	251	0	0	246
		Val	91.45	91.45	91.39	91.42	65	6	7	74
		Test	**93.55**	**93.61**	**93.57**	**93.59**	54	7	7	90
	DistilBERT	Train	100	100	100	100	251	0	0	246
		Val	88.82	88.98	89.03	89.00	67	12	5	68
		Test	90.32	90.33	90.33	90.33	42	4	5	42
4MLP	BERT	Train	100	100	100	100	251	0	0	246
		Val	90.13	90.17	90.28	90.23	67	10	5	70
		Test	91.40	91.40	91.40	91.40	43	4	4	42
	RoBERTa	Train	100	100	100	100	251	0	0	246
		Val	91.45	91.45	91.39	91.42	65	6	7	74
		Test	*92.47*	*92.62*	*92.51*	*92.57*	42	2	5	44
	DistilBERT	Train	100	100	100	100	251	0	0	246
		Val	90.13	90.09		90.11	65	8	7	72
		Test	87.10	87.35	87.14	87.25	39	4	8	42
10L-ResNet-CNN	BERT	Train	100	100	100	100	251	0	0	246
		Val	90.79	91.25	91.11	91.18	70	12	2	68
		Test	90.32	90.35	90.31	90.33	43	5	4	41
	RoBERTa	Train	100	100	100	100	251	0	0	246
		Val	91.45	91.45	91.39	91.42	65	6	7	74
		Test	93.55	93.61	93.57	93.59	53	7	8	90
	DistilBERT	Train	100	100	100	100	251	0	0	246
		Val	88.82	88.98	89.03	89.00	67	12	5	68
		Test	91.40	91.40	91.40	91.40	43	4	4	42

A Accuracy, *P* precision, *R* recall, *F1* F1 score, *TP* true positive, *FP* false positive, *FN* false negative, *TN* true negative, *SOTA* state-of-the-art

Bold value represent best results, where as underlined and italicized represent the second-best and third-best results in tables

performed better when paired with RoBERTa embedding. However, the differences in performance, when compared to the models utilising BERT and DistilBERT embedding, are not significant.

4.2 Comparison of results with the SOTA on true/false datasets

For the classification of true and false rumours, Tables 8, 9 and 10 and Figs. 6, 7 and 8 present the performance of all proposed models on the Pheme T/F, Twitter 15 T/F, and Twitter 16 T/F datasets. Table 8 presents the

Table 10 Comparison of the proposed method with state-of-the-art techniques for Twitter 16 T/F dataset

Classifier models			Best results on Twitter 15 T/F (410 tweets; 205 T, 205 F)				Confusion matrix			
SOTA models			A (%)	P (%)	R (%)	F1 (%)	TP	FP	FN	TN
Gcan [24]	70:30 for train:test		90.84	75.94	76.32	75.93	–	–	–	–
Pelrine et al.[58]	70:10:20 for train:val:test		–	–	–	95.7 ±2.8	–	–	–	–
Proposed Models [10%: testing, 75% and 25% of rest of the data for training and validation, respectively]										
2MLP	BERT	Train	99.27	99.27	99.29	99.28	134	2	0	139
		Val	85.71	85.69	85.69	85.69	44	7	7	40
		Test	97.44	97.50	97.50	97.50	19	0	1	19
	RoBERTa	Train	100	100	100	100	134	0	0	141
		Val	90.82	90.79	90.84	90.82	46	4	5	43
		Test	92.31	92.37	92.37	92.37	18	1	2	18
	7emDistilBERT	Train	100	100	100	100	134	0	0	141
		Val	91.84	92.22	91.66	91.94	49	6	2	41
		Test	100	100	100	100	20	0	0	19
4MLP	BERT	Train	99.27	99.27	99.29	99.28	134	2	0	139
		Val	85.71	85.69	85.69	85.69	44	7	7	40
		Test	*97.44*	*97.50*	*97.50*	*97.50*	19	0	1	19
	RoBERTa	Train	100	100	100	100	134	0	0	141
		Val	91.84	91.84	91.91	91.87	46	3	5	44
		Test	94.87	95.24	95.00	95.12	18	0	2	19
	DistilBERT	Train	100	100	100	100	134	0	0	141
		Val	91.84	92.22	91.66	91.94	49	6	2	41
		Test	**100**	**100**	**100**	**100**	20	0	0	19
10L-ResNet-CNN	BERT	Train	94.18	94.67	94.33	94.50	134	16	0	125
		Val	90.82	92.50	90.43	91.45	51	9	0	38
		Test	92.31	93.48	92.11	92.79	20	3	0	16
	RoBERTa	Train	100	100	100	100	134	0	0	141
		Val	91.84	91.84	91.91	91.87	46	3	5	44
		Test	94.87	95.24	95.00	95.12	18	0	2	19
	DistilBERT	Train	99.64	99.63	99.65	99.64	134	1	0	140
		Val	91.84	93.22	91.49	92.35	51	8	0	39
		Test	94.87	95.46	94.74	95.10	20	2	0	17

A Accuracy, *P* precision, *R* recall, *F1* F1 score, *TP* true positive, *FP* false positive, *FN* false negative, *TN* true negative, *SOTA* state-of-the-art

Bold value represent best results, where as underlined and italicized represent the second-best and third-best results in tables

experimental results of the proposed models on the Pheme T/F dataset which categorised as small and imbalanced labels (refer to Table 1). Notably, the table illustrates that all the proposed models surpass the state-of-the-art model by Wu et al. [51], achieving improvements ranging from 5% to 8% across all metrics. While the proposed models may not attain superior F1 scores, they deliver results that are closely comparable to those reported by Pelrine et al.[58].

Furthermore, Tables 8, 9 and 10 highlight the consistent superiority of RoBERTa embeddings over BERT and DistillBERT across all classifier models. Interestingly, despite its simplicity, the MLP model achieved slightly better results than the more complex ResNet-CNN. This variance in performance could be attributed to the fact that ResNet-CNN is primarily designed for image processing and may not be optimally suited for text data. Additionally, it is possible that the hyperparameters for the ResNet-CNN model were not well optimised for the specific text classification task, contributing to its lower performance.

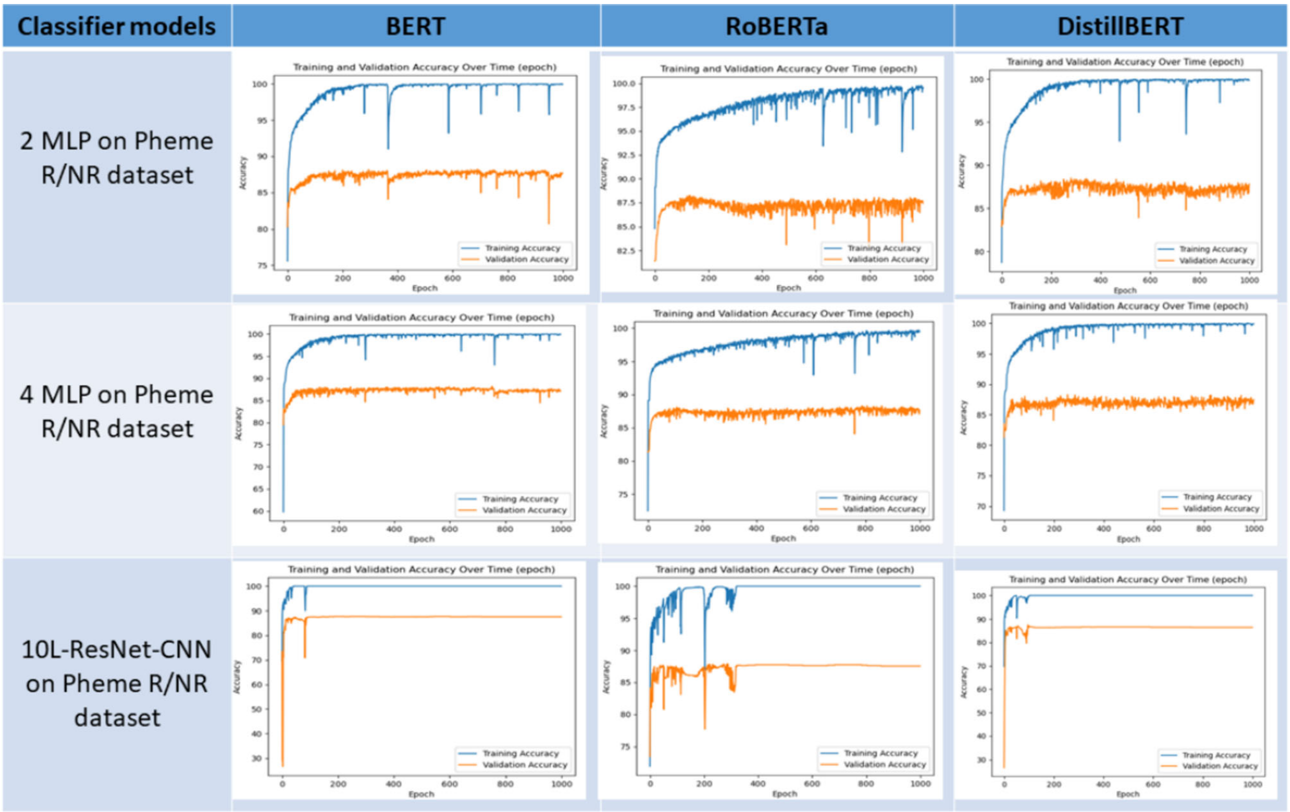


Fig. 3 The accuracy graphs on Pheme R/NR dataset

However, concern remains: despite the proposed models showing significant improvements over the SOTA, the accuracy graph in Fig. 6 suggests potential overfitting, with some models showing a training–validation gap exceeding 10%. This disparity may be due to the dataset's small and imbalanced nature. Table 8 and Fig. 6 show that all proposed models paired with RoBERTa embeddings have smaller training–validation gaps compared to those using BERT and DistilBERT.

Table 9 along with Fig. 7 and Table 10 with Fig. 8 compare the performance of the proposed models in classifying true and false rumours on the Twitter 15 T/F and Twitter 16 T/F datasets, respectively. Both datasets have smaller data sizes and relatively balanced label distributions (refer to Table 1). From these Tables, we can see that all the proposed models achieve high performance and outperform the state-of-the-art model by Lu et al. [24]. Furthermore, they show F1 scores comparable to those in Pelrine et al. [58], which, unfortunately, only reported the F1 score, making it difficult to compare other performance metrics.

In general, across the Twitter 15 T/F dataset, all proposed models exhibit improved performance when integrated with text embeddings generated by RoBERTa, the largest language model. Conversely, on the Twitter 16 T/F dataset, the highest performance was observed when combining the 4MLP model with text embeddings generated from the smallest language model DistilBERT, followed by the 4MLP model paired with BERT embeddings and the 2MLP model with BERT embeddings. Given Twitter 16 T/F is considered a small and balanced dataset (refer to Table 1), we observed that this difference can be attributed to the theory that smaller models with fewer parameters are less likely to memorise noise in the data, leading to better generalisation on limited samples. While disparities between training and validation results on some models might indicate potential overfitting issues, it is worth noting that some are under 10%, which is considered normal for small datasets [67].

Similarly, in the classification of rumours and non-rumours tweets, we addressed the small dataset issue by augmenting the dataset by combining the Pheme T/F, Twitter 15 T/F, and Twitter 16 T/F datasets into a single

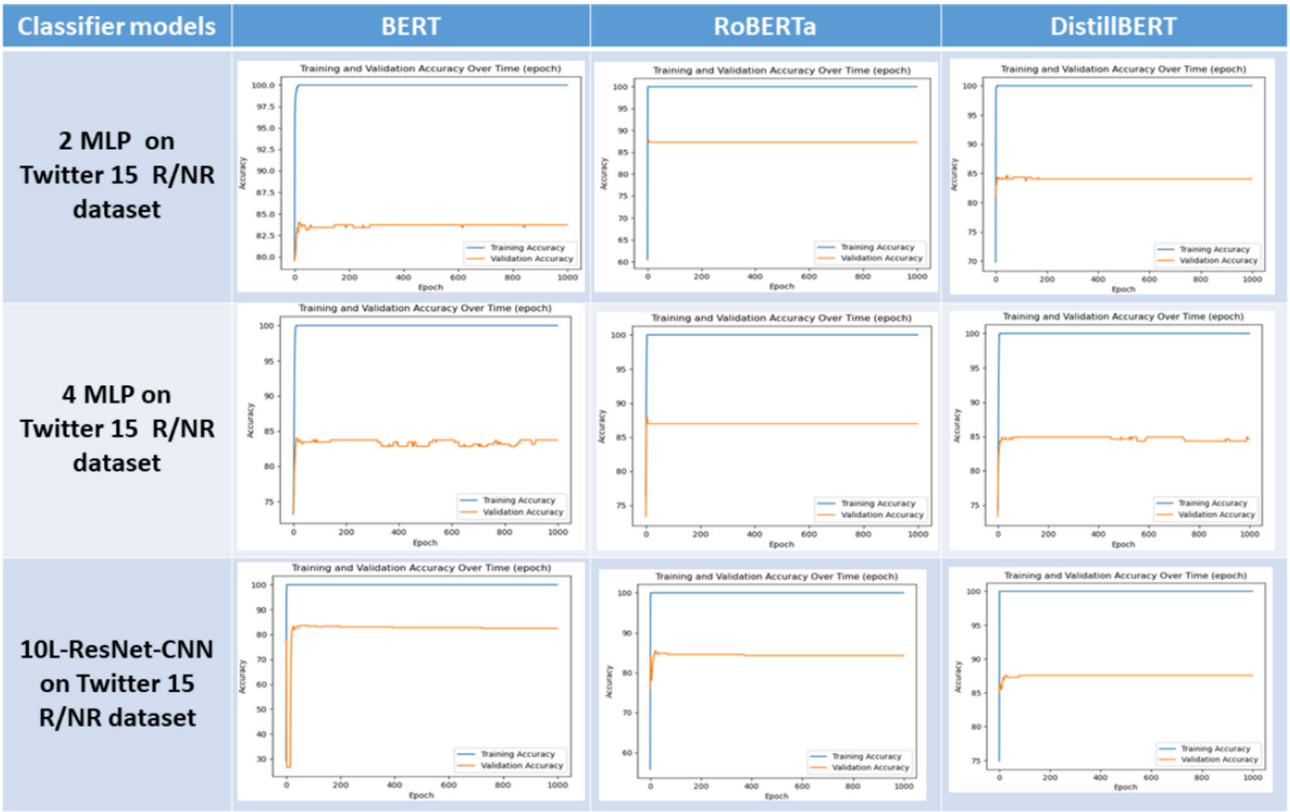


Fig. 4 The accuracy graphs on Twitter 15 R/NR dataset

dataset referred to as “Combined T/F” (see Table 1). We trained all the proposed models on this augmented dataset, and the results are presented in Table 11. This table illustrates that all the proposed models, when coupled with text embeddings from RoBERTa, BERT, and DistillBERT, achieve excellent performance with minimal differences in results.

Additionally, we also performed fivefold cross-validation for 4MLP and 10ResNet-CNN, which achieved higher performance on Pheme T/F and Twitter T/F datasets, to evaluate their generalisation capabilities. We present their cross-validation results in Table 12. It can be seen that the proposed models consistently perform well across the five folds. While there are slight variations in results, this is common during cross-validation and may be due to the limited data available for each fold. From these indications, we can conclude that the proposed models have the generalisation capability to classify rumours on Twitter.

4.3 Ablation study

We conducted an ablation study to evaluate the impact of the fine-tuning process and the necessity of the additional neural network training phase in our proposed architecture. Specifically, we compared the performance of the proposed model with a baseline model without fine-tuning, as well as a model that only fine-tunes the BERT-based models for classification.

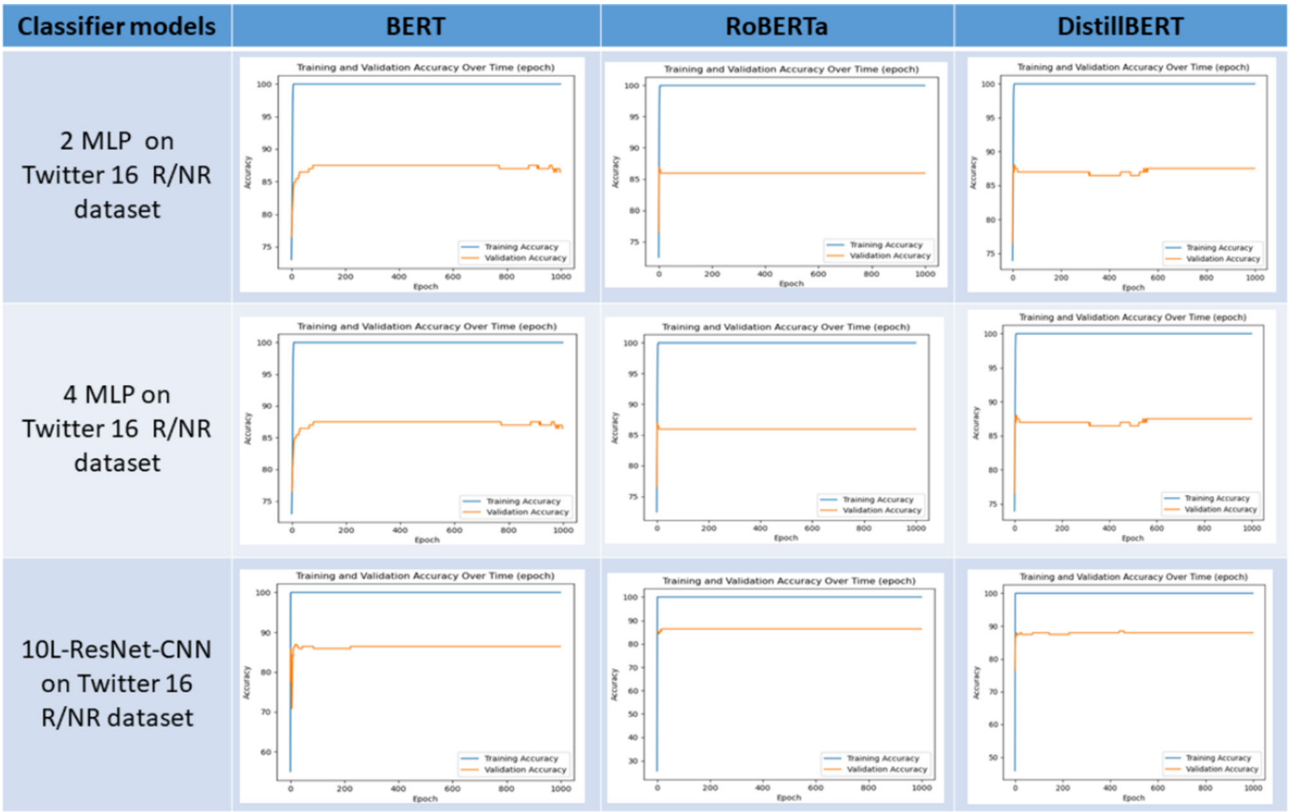


Fig. 5 The accuracy graphs on Twitter 16 R/NR dataset

4.3.1 Experimental setup for ablation study

The datasets and experimental setup used in this ablation study are consistent with those described in the Materials and Methods section. We evaluated three different models: the baseline model, the fine-tuned model, and the proposed model. The baseline model refers to the BERT-based language model (BERT, RoBERTa, or DistilBERT) used directly for predictions without fine-tuning. The fine-tuned model is the baseline model fine-tuned on the dataset, also used directly for making predictions. The proposed model is a fine-tuned model with an additional training phase for a neural network classifier. In this ablation study, we used a two-layer multilayer perceptron (2MLP) as the classifier, as outlined in our proposed architecture.

4.3.2 Results and analysis of the ablation study

The results of the ablation study are summarised in Table 13, which compares the performance of three different models- the baseline model (unfine-tuned), the fine-tuned model, and the proposed model-across three datasets (Twitter 15, Twitter 16, and Pheme2) for two tasks: rumour/non-rumour classification and true/false classification.

Table 13 clearly shows that fine-tuning the targeted datasets leads to superior performance compared to the baseline model without fine-tuning. All fine-tuned BERT, RoBERTa, and DistilBERT consistently outperformed their respective baseline models. Across all datasets and tasks, there is a consistent and significant improvement in performance when moving from the unfine-tuned baseline model to the fine-tuned model. This indicates that fine-tuning the BERT-based language models on specific datasets substantially enhances their ability to make accurate predictions.

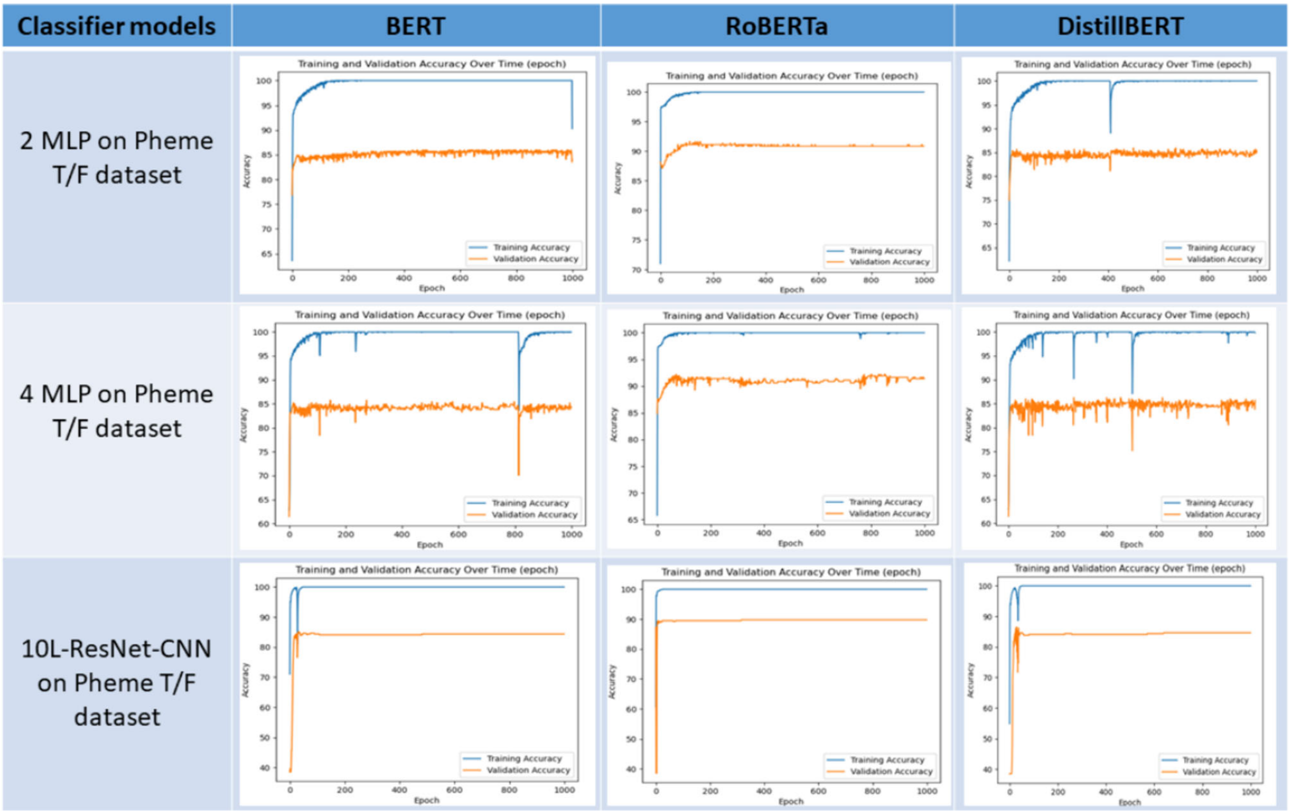


Fig. 6 The accuracy graphs on Pheme T/F dataset

Additionally, the choice of a neural network classifier plays a significant role in the architecture of the proposed model. The proposed model, which includes an additional neural network classifier trained on top of the fine-tuned transformer model, consistently outperforms both the unfine-tuned and fine-tuned models in nearly all cases. The improvement in accuracy and F1 score across all datasets suggests that the additional complexity of the proposed model is justified, as it provides more accurate and reliable predictions than simply fine-tuning the transformer model.

The analysis of the experimental results confirms that the proposed model, which includes both fine-tuning and an additional neural network training phase, significantly outperforms both the unfine-tuned baseline model and the fine-tuned model alone. This demonstrates that the added complexity of the proposed architecture is indeed beneficial and necessary for achieving higher accuracy and better overall performance in rumour detection tasks across various datasets.

4.4 Enhancing fine-tuning process and adjusting the architecture of the proposed model

Given 4MLP classifier model exhibits surpassing performance over other proposed models but at the same time also reveals signs indicating an overfitting issue. We focus on the 4MLP model to be improved to address the revealed issue. As seen in Figs. 3 and 6, the accuracy curve of the 4MLP model on Pheme R/NR and Pheme T/F shows a ripple and some dips in the training and validation set which may indicate an issue. Since the ripple issue occurs only in the Pheme R/NR and Pheme T/F datasets, we speculate that this problem may be attributed to the characteristics of the Pheme dataset, which is small and unbalanced. Another possible reason could be suboptimal hyperparameter settings and the complexity of the MLP architecture, which might make it prone to overfitting.

To address these issues, we made several adjustments, including reducing the composition of neurons and fine-tuning various parameters such as learning rate, batch size, and the number of epochs. We experimented with

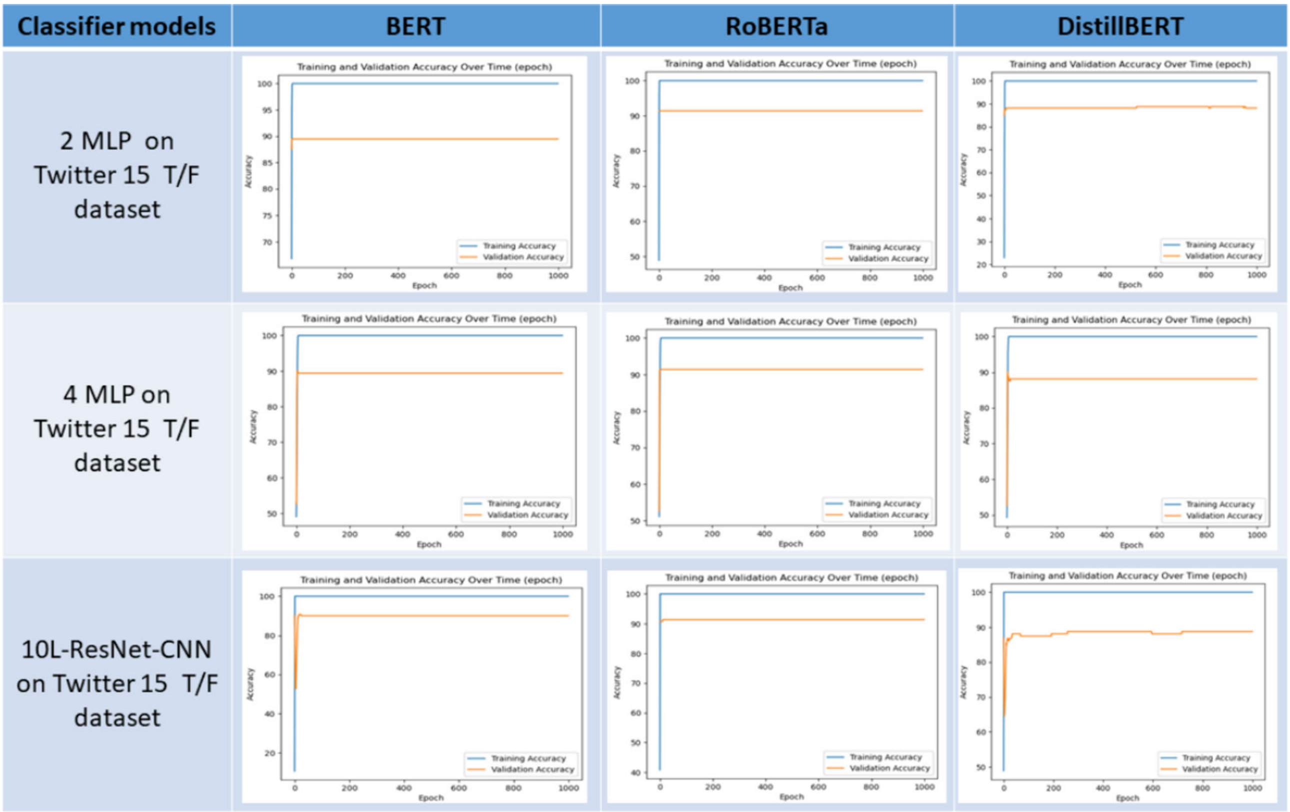


Fig. 7 The accuracy graphs on Twitter 15 T/F dataset

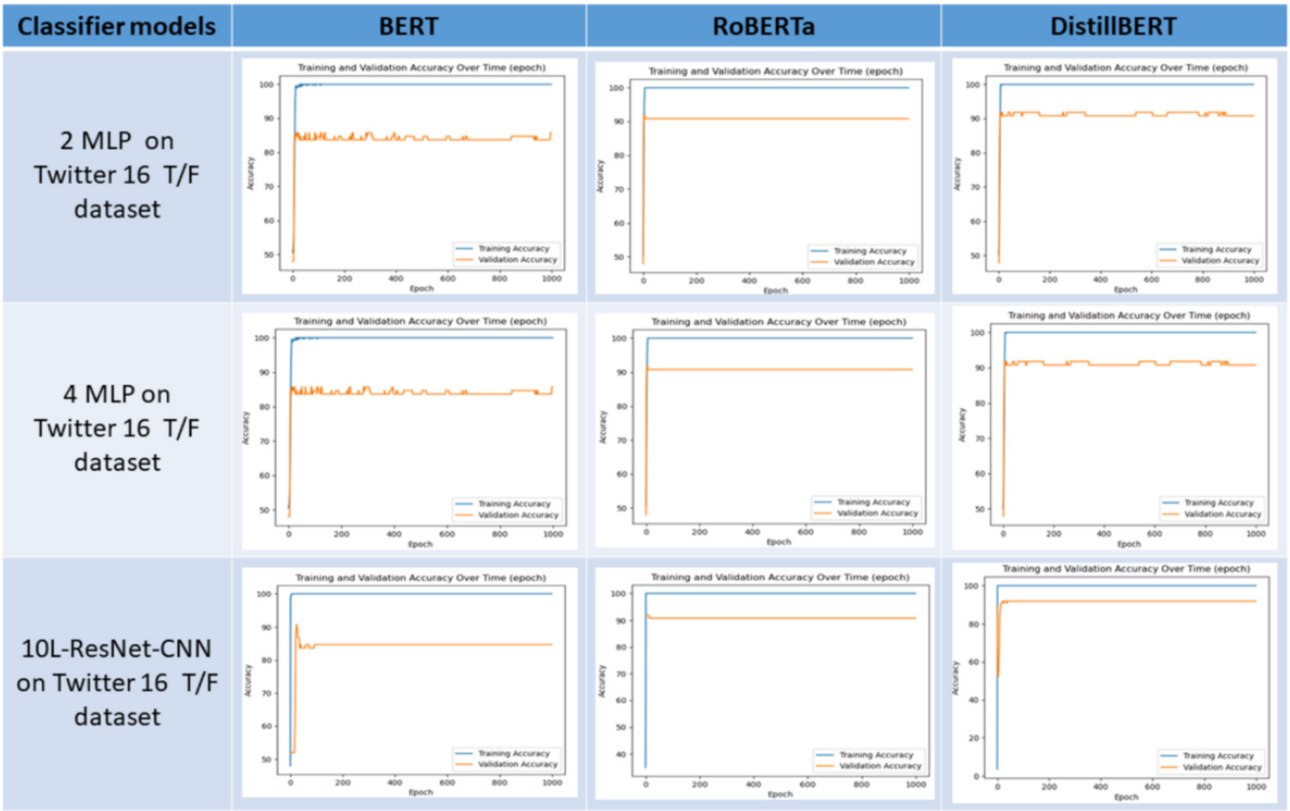


Fig. 8 The accuracy graphs on Twitter 16 T/F dataset

Table 11 Comparison of the proposed method with state-of-the-art techniques for combined T/F dataset

Classifier models			Best results on Combined T/F (2859 tweets; 1646 T, 1213 F)				Confusion matrix			
			A (%)	P (%)	R (%)	F1 (%)	TP	FP	FN	TN
Proposed Models [10%: testing, 75% and 25% of rest of the data for training and validation, respectively]										
2MLP	BERT	Train	99.95	99.95	99.95	99.95	951	1	0	996
		Val	97.26	97.16	97.35	97.25	274	12	5	330
		Test	96.90	96.91	96.87	96.89	132	4	5	149
	RoBERTa	Train	99.54	99.55	99.53	99.54	942	0	9	997
		Val	99.03	99.10	98.96	99.03	274	1	5	341
		Test	*97.59*	*97.65*	*97.52*	*97.59*	132	2	5	151
	DistilBERT	Train	100	100	100	100	951	0	0	997
		Val	98.39	98.34	98.41	98.38	275	6	4	336
		Test	97.24	97.28	97.20	97.24	132	3	5	150
4MLP	BERT	Train	100	100	100	100	951	0	0	997
		Val	97.26	97.16	97.35	97.25	274	12	5	330
		Test	96.90	96.91	96.87	96.89	132	4	5	149
	RoBERTa	Train	99.54	99.55	99.53	99.54	942	0	9	997
		Val	99.03	99.10	98.96	99.03	274	1	5	341
		Test	97.59	97.65	97.52	97.59	132	2	5	151
	DistilBERT	Train	100	100	100	100	951	0	0	997
		Val	98.55	98.49	98.59	98.54	276	6	3	336
		Test	97.24	97.28	97.20	97.24	132	3	5	150
10L-ResNet-CNN	BERT	Train	99.69	99.70	99.69	99.69	945	0	6	997
		Val	97.10	97.04	97.10	97.07	271	10	8	332
		Test	96.21	96.21	96.18	96.20	131	5	6	148
	RoBERTa	Train	99.54	99.55	99.53	99.54	942	0	9	997
		Val	98.39	98.53	98.24	98.39	270	1	9	341
		Test	**97.59**	**97.72**	**97.48**	**97.60**	131	1	6	152
	DistilBERT	Train	100	100	100	100	951	0	0	997
		Val	97.42	97.29	97.60	97.44	277	14	2	328
		Test	95.86	95.83	95.89	95.86	132	7	5	146

Bold value represent best results, where as underlined and italicized represent the second-best and third-best results in tables

different combinations of reduced neuron compositions and adjusted parameter settings to find the optimal configuration and present the result in Figs. 9 and 10. Ultimately, we obtained the best results for Pheme R/NR using a 4MLP with a neuron composition of 256-128-64-32. We set the learning rate to 9e–5, the batch size to 512, and the maximum number of epochs to 1000. For Pheme T/F, we achieved the best results using the same 4MLP architecture with a learning rate of 5e-5.

Additionally, we enhanced the quality of embeddings for all BERT-based models by increasing the number of training epochs from 20 to 40. Unfortunately, due to limited computational resources and GPU memory constraints, we could not further extend the number of epochs. The results of these improvements are presented in Table 14, Figs. 11 and 12. The experiment's results demonstrate that reducing the size of MLP and adjusting the parameters setting is needed to obtain a better neural networks model. Referring to Figs. 3, 6, 9, and 10, the ripples in curves of accuracy on the 4MLP models were reduced after reducing the size of MLP and the learning rate. We note that the optimum hyperparameter setting depends on the characteristics of the dataset and neural networks model; hence, a different task, models, and dataset would need a slightly tailored architecture and parameter settings. Additionally, we also observed that there is no significant improvement in text embedding obtained by increasing the number of fine-tuning epochs from 20 to 40 on BERT-based language models

Table 12 Cross-validation evaluation of the proposed models on true-/false-rumour dataset

Dataset	Classifier models		Fivefold Cross-validation on R/NR datasets				
	Classifier	Embedding	Fold	A (%)	P (%)	R (%)	F1 (%)
Pheme T/F	4MLP	RoBERTa	Fold-1	88.61	88.41	87.38	87.89
			Fold-2	89.24	89.25	87.89	88.57
			Fold-3	89.87	89.32	89.32	89.32
			Fold-4	89.87	89.79	88.71	89.25
			Fold-5	89.87	89.32	89.32	89.32
			Avg	89.49	89.22	88.52	88.87
	10L-ResNet-CNN	RoBERTa	Fold-1	88.61	87.98	87.98	87.98
			Fold-2	88.61	87.85	88.29	88.07
			Fold-3	89.24	88.58	88.80	88.69
			Fold-4	87.98	87.25	87.47	87.36
			Fold-5	90.51	90.09	89.83	89.96
			Avg	88.99	88.35	88.48	88.41
Twitter T/F	4MLP	RoBERTa	Fold-1	84.95	88.33	85.11	86.69
			Fold-2	91.40	91.67	91.44	91.56
			Fold-3	92.47	92.63	92.51	92.57
			Fold-4	93.55	94.23	93.62	93.92
			Fold-5	92.47	93.40	92.55	92.97
			Avg	90.97	92.05	91.05	91.54
	10L-ResNet-CNN	RoBERTa	Fold-1	91.40	92.05	91.47	91.76
			Fold-2	92.47	92.93	92.53	92.73
			Fold-3	89.25	91.07	89.36	90.21
			Fold-4	89.25	91.07	89.36	90.21
			Fold-5	93.55	93.61	93.57	93.59
			Avg	91.18	92.15	91.26	91.70

(compare results in Tables 3, 8 and 14). This experimental result aligns with the findings of a previous study [62], which recommended an optimum iteration of 20 epochs for fine-tuning transformer language models. However, the study also suggested that longer fine-tuning may contribute to the stability of the model.

Our study gained theoretical contributions by demonstrating the effectiveness of our rumour detection models, which outperform existing approaches. We also provide theoretical insights into text embeddings by comparing the quality of vectors generated by different BERT-based models. These findings have implications for both the rumour detection and natural language processing communities.

4.5 Limitations

Our study has provided valuable insights into the effectiveness of our proposed models for rumour detection on Twitter and a comparison of the quality of text embedding generated from the fine-tuned various BERT-based language models. However, it is important to acknowledge its possible limitations. Firstly, we experimented with limited computational resources, utilising only an 8GB GPU. This computational constraint hindered our flexibility in fine-tuning parameters and limited the maximum number of epochs.

Secondly, the datasets used in this study, while carefully curated, are relatively medium in size. This limitation may restrict the generalisability of our findings to very large and more diverse datasets. Moreover, the imbalanced nature of some datasets, particularly Pheme R/NR, Pheme T/F, Twitter 15 R/NR, and Twitter 16 R/NR datasets, could introduce bias into the model's performance evaluation.

Table 13 Comparison results of the ablation study

Datasets	Transformer models	Configuration models	Rumour/non-rumour				True/false			
			A	P	R	F1	A	P	R	F1
PHEME	BERT	Baseline model	0.780	0.758	0.746	0.752	0.765	0.764	0.751	0.757
		Fine-tuned model	0.849	0.837	0.816	0.826	0.835	0.834	0.821	0.827
		Proposed model-2MLP	0.892	0.867	0.849	0.858	0.880	0.879	0.866	0.872
	RoBERTa	Baseline model	0.784	0.747	0.767	0.757	0.803	0.801	0.800	0.800
		Fine-tuned model	0.864	0.827	0.847	0.837	0.883	0.881	0.880	0.880
		Proposed model-2MLP	0.892	0.855	0.875	0.876	0.911	0.907	0.907	0.907
	DistilBERT	Baseline model	0.764	0.725	0.729	0.727	0.757	0.751	0.746	0.748
		Fine-tuned model	0.841	0.802	0.806	0.804	0.834	0.828	0.823	0.825
		Proposed model-2MLP	0.868	0.829	0.833	0.831	0.861	0.855	0.850	0.852
Twitter 15	BERT	Baseline model	0.674	0.670	0.663	0.667	0.700	0.713	0.717	0.715
		Fine-tuned model	0.828	0.833	0.825	0.829	0.862	0.875	0.879	0.877
		Proposed model-2MLP	0.883	0.890	78.001	0.831	0.914	0.914	0.914	0.914
	RoBERTa	Baseline model	0.703	0.675	0.641	0.658	0.747	0.755	0.752	0.753
		Fine-tuned model	0.842	0.814	0.780	0.797	0.886	0.897	0.893	0.895
		Proposed model-2MLP	0.891	0.863	0.829	0.846	0.935	0.936	0.936	0.936
	DistilBERT	Baseline model	0.669	0.627	0.605	0.616	0.705	0.705	0.705	0.705
		Fine-tuned model	0.828	0.786	0.764	0.775	0.864	0.864	0.864	0.864
		Proposed model-2MLP	0.867	0.825	0.803	0.814	0.903	0.903	0.903	0.903
Twitter 16	BERT	Baseline model	0.647	0.605	0.622	0.613	0.793	0.795	0.793	0.794
		Fine-tuned model	0.790	0.805	0.789	0.797	0.961	0.963	0.961	0.962
		Proposed model-2MLP	0.802	0.779	0.734	0.699	0.974	0.975	0.975	0.975
	RoBERTa	Baseline model	0.738	0.762	0.675	0.716	0.718	0.717	0.716	0.716
		Fine-tuned model	0.898	0.922	0.835	0.876	0.878	0.877	0.876	0.876
		Proposed model-2MLP	0.938	0.962	0.875	0.916	0.923	0.924	0.924	0.924
	DistilBERT	Baseline model	0.651	0.607	0.541	0.572	0.826	0.827	0.826	0.826
		Fine-tuned model	0.790	0.746	0.680	0.711	0.965	0.966	0.965	0.965
		Proposed model-2MLP	0.827	0.783	0.717	0.748	1.000	1.000	1.000	1.000

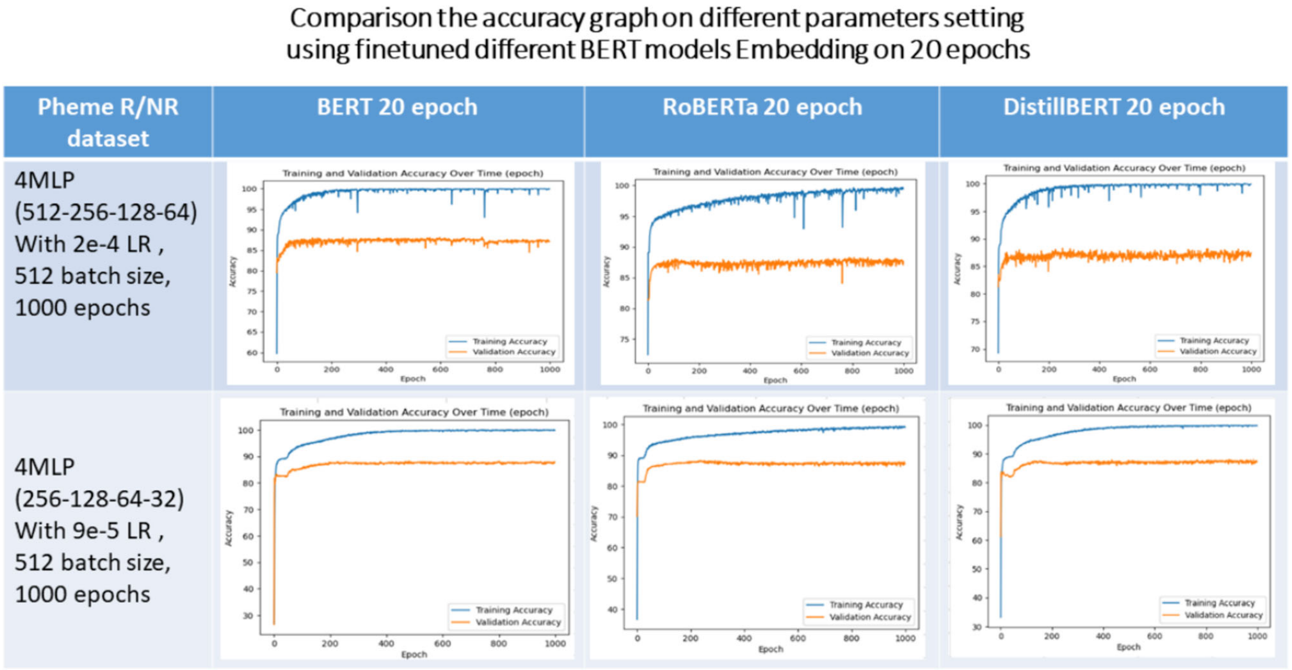
A Accuracy, *P* precision, *R* recall, *F1* F1 score

Thirdly, a comprehensive comparison with the state-of-the-art models is hindered by their omission of performance results on training and validation data in their research studies. Hence, although we provide a clear and detailed evaluation of training, validation, and testing results, a full comparison with SOTA models is not possible. As a result, our model's comparison with SOTA models remains limited.

Furthermore, our study focussed primarily on English-language tweets only, which might not capture the full spectrum of languages and cultural contexts present on social media platforms. Another limitation to consider is the absence of a real-time evaluation scenario, as our experiments were conducted on static datasets. Lastly, while we conducted hyperparameter tuning, model selection, and performance evaluation rigorously, the field of deep learning is continually evolving, and future improvements in model architectures and techniques may obtain better results.

4.6 Future directions

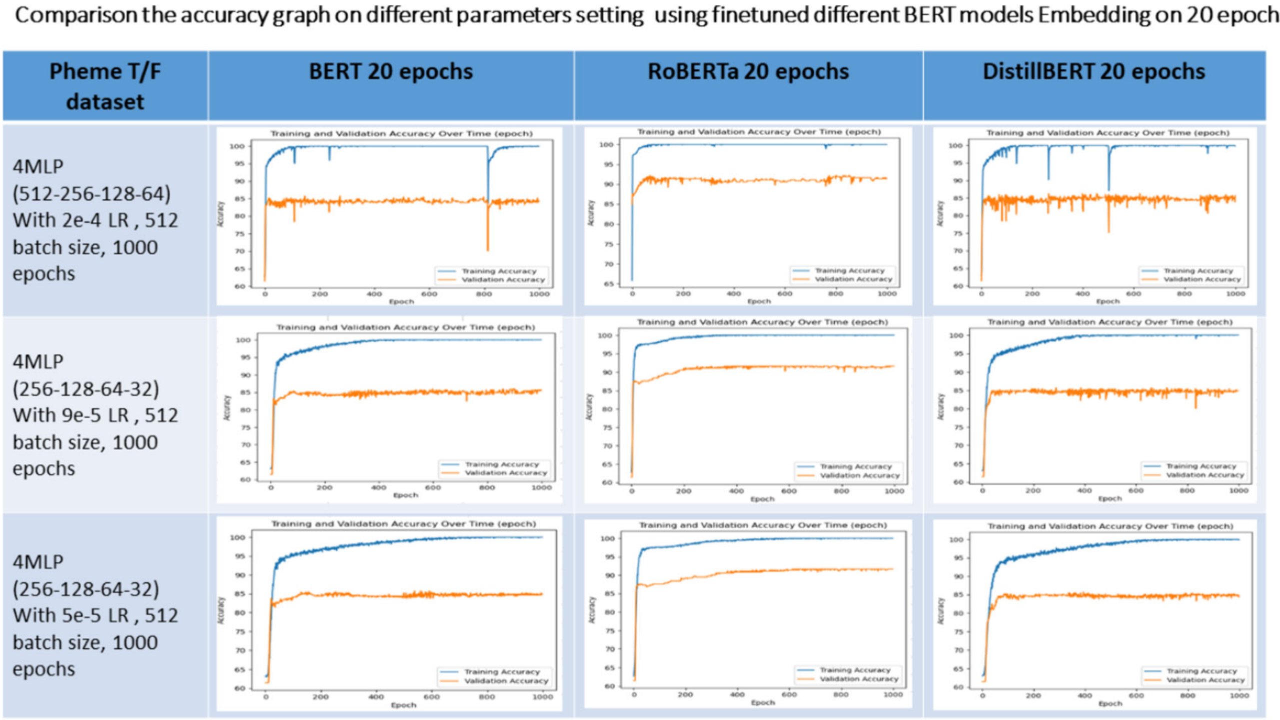
This study has highlighted key aspects of rumour detection using deep learning and BERT-based text embeddings, while also opening several paths for future research. Firstly, broadening our investigation to encompass a



Comparison the accuracy graph on different parameters setting using finetuned different BERT models Embedding on 20 epochs

Pheme R/NR dataset	BERT 20 epoch	RoBERTa 20 epoch	DistillBERT 20 epoch
4MLP (512-256-128-64) With 2e-4 LR , 512 batch size, 1000 epochs			
4MLP (256-128-64-32) With 9e-5 LR , 512 batch size, 1000 epochs			

Fig. 9 The improved 4MLP's accuracy graphs using different parameters setting on Pheme R/NR datasets



Comparison the accuracy graph on different parameters setting using finetuned different BERT models Embedding on 20 epochs

Pheme T/F dataset	BERT 20 epochs	RoBERTa 20 epochs	DistillBERT 20 epochs
4MLP (512-256-128-64) With 2e-4 LR , 512 batch size, 1000 epochs			
4MLP (256-128-64-32) With 9e-5 LR , 512 batch size, 1000 epochs			
4MLP (256-128-64-32) With 5e-5 LR , 512 batch size, 1000 epochs			

Fig. 10 The improved 4MLP's accuracy graphs using different parameters setting on Pheme T/F datasets

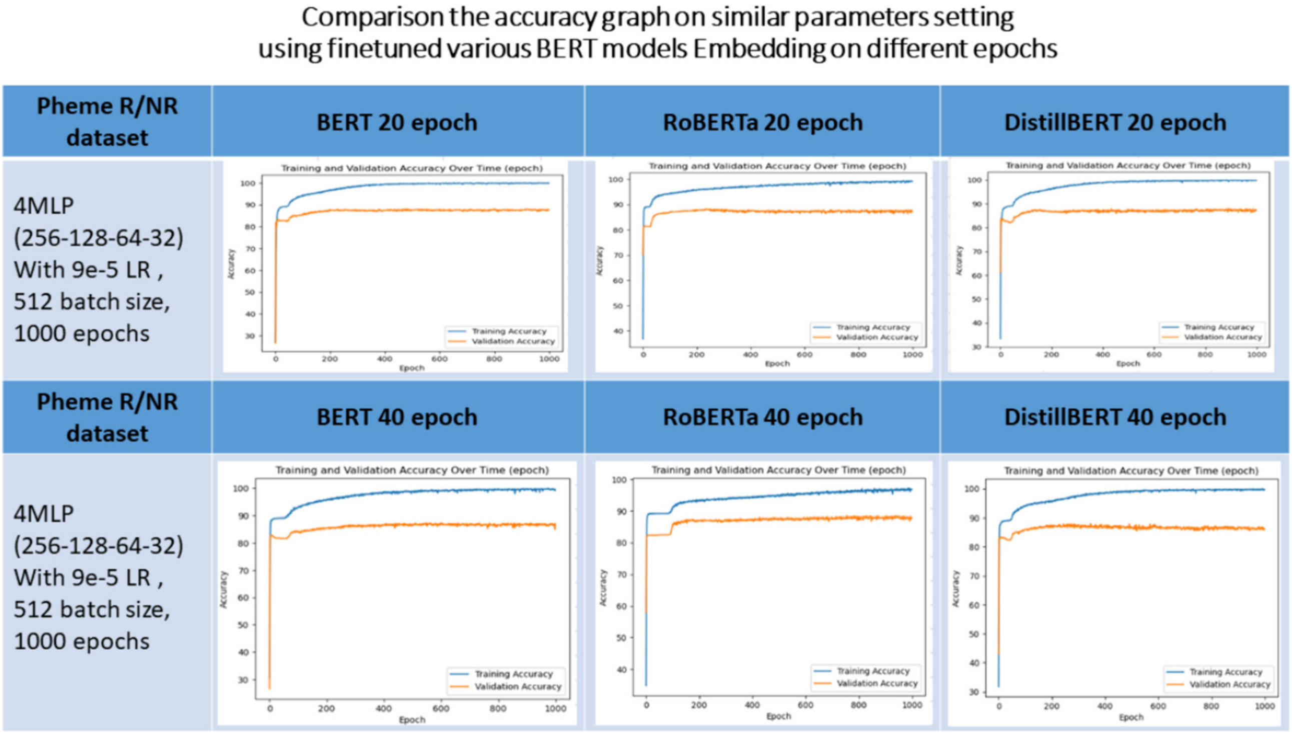
wider array of languages and social media platforms can provide a more comprehensive understanding of how rumours spread across diverse online ecosystems. Furthermore, exploring transfer learning techniques, such as domain adaptation and multilingual models, can enhance the applicability of our models to different languages and cultural contexts. Additionally, incorporating real-time data streams and dynamic model updates could create more resilient systems for rumour detection in fast-changing online environments. Future research efforts could

Table 14 Comparison of the improved proposed method on various fine-tuned BERT-based embeddings on 20 and 40 epochs

Dataset	Classifier models			Best results on Combined R/NR (6425 tweets; 2402 R, 4023 NR)				Confusion matrix			
	Model	Embeddings	Set	A (%)	P (%)	R (%)	F1	TP	FP	FN	TN
Pheme R/NR	4MLP (LR 9e-5)	BERT (fine-tuned 20 epochs)	Train	99.977	99.957	99.984	1.00	3184	0	1	1151
			Val	88.235	85.51	83.679	0.85	1003	101	71	287
			Test	88.836	86.761	83.734	0.85	436	45	25	121
		RoBERTa (fine-tuned 20 epochs)	Train	96.287	94.117	96.945	0.96	3043	19	142	1132
			Val	88.304	84.358	87.594	0.86	957	54	117	334
			Test	88.836	85.201	87.011	0.86	419	28	42	138
		DistilBERT (fine-tuned 20 epochs)	Train	99.585	99.23	99.717	0.99	3167	0	18	1151
			Val	87.962	85.332	82.999	0.84	1005	107	69	281
			Test	86.922	83.763	81.855	0.83	427	48	34	118
Pheme R/NR	4MLP (LR 9e-5)	BERT (fine-tuned 40 epochs)	Train	99.585	99.308	99.634	0.99	3170	3	15	1148
			Val	87.346	83.945	83.321	0.84	987	98	87	290
			Test	88.357	85.349	84.373	0.85	428	40	33	126
		RoBERTa (fine-tuned 40 epochs)	Train	97.486	95.975	97.817	0.97	3093	17	92	1134
			Val	88.44	84.78	86.37	0.86	975	70	99	318
			Test	88.517	84.974	86.023	0.85	421	32	40	134
		DistilBERT (fine-tuned 40 epochs)	Train	98.593	97.635	98.848	0.98	3131	7	54	1144
			Val	88.03	84.803	84.28	0.85	991	92	83	296
			Test	86.922	83.269	83.011	0.83	421	42	40	124
Pheme T/F	4MLP (LR 5e-5)	BERT (fine-tuned 20 epochs)	Train	99.49	99.5	99.404	0.99	430	2	4	740
			Val	85.984	85.332	84.947	0.85	115	24	28	204
			Test	88.608	88.414	87.375	0.88	50	7	11	90
		RoBERTa (fine-tuned 20 epochs)	Train	99.575	99.616	99.472	1.00	430	1	4	741
			Val	91.644	91.676	90.595	0.91	123	11	20	217
			Test	91.139	90.654	90.654	0.91	54	7	7	90
		DistilBERT (fine-tuned 20 epochs)	Train	99.405	99.432	99.289	0.99	429	2	5	740
			Val	85.714	85.011	84.727	0.85	115	25	28	203
			Test	85.443	84.711	84.494	0.85	49	11	12	86
Pheme T/F	4MLP (LR 5e-5)	BERT (fine-tuned 40 epochs)	Train	99.8	99.7	99.8	99.7	433	1	1	741
			Val	85.714	85.251	84.336	0.85	112	22	31	206
			Test	87.975	88.194	86.251	0.87	48	6	13	91
		RoBERTa (fine-tuned 40 epochs)	Train	99.82	99.79	99.86	99.79	433	1	1	741
			Val	87.871	88.118	86.091	0.87	112	14	31	214
			Test	88.608	88.72	87.071	0.88	49	6	12	91
		DistilBERT (fine-tuned 40 epochs)	Train	99.915	99.933	99.885	99.96	432	1	1	742
			Val	85.175	84.331	84.419	0.84	116	28	27	200
			Test	80.38	79.284	79.457	0.79	46	16	15	81

A Accuracy, *P* precision, *R* recall, *F1* F1 score, *TP* true positive, *FP* false positive, *FN* false negative, *TN* true negative, *SOTA* state-of-the-art

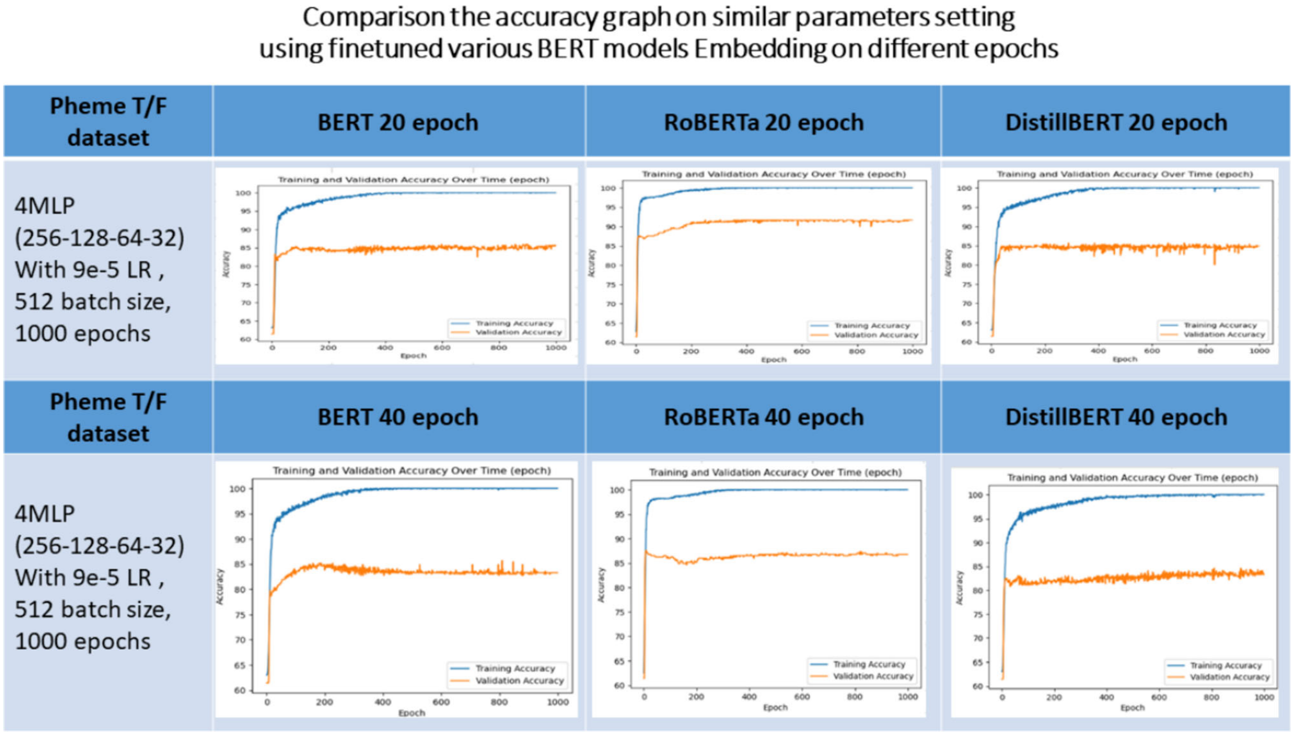
also delve deeper into improving the explainability and interpretability of model predictions, addressing the "black-box" nature of deep learning models to make them more transparent and accountable. Since BERT-based models have high computational costs that limit their usability in resource-constrained settings, future research should focus on developing a concise yet efficient BERT model to address this computational challenge. By



Comparison the accuracy graph on similar parameters setting
using finetuned various BERT models Embedding on different epochs

Pheme R/NR dataset	BERT 20 epoch	RoBERTa 20 epoch	DistillBERT 20 epoch
4MLP (256-128-64-32) With 9e-5 LR , 512 batch size, 1000 epochs			
Pheme R/NR dataset	**BERT 40 epoch**	**RoBERTa 40 epoch**	**DistillBERT 40 epoch**
4MLP (256-128-64-32) With 9e-5 LR , 512 batch size, 1000 epochs			

Fig. 11 The improved 4MLP's accuracy graphs on Pheme R/NR dataset



Comparison the accuracy graph on similar parameters setting
using finetuned various BERT models Embedding on different epochs

Pheme T/F dataset	BERT 20 epoch	RoBERTa 20 epoch	DistillBERT 20 epoch
4MLP (256-128-64-32) With 9e-5 LR , 512 batch size, 1000 epochs			
Pheme T/F dataset	**BERT 40 epoch**	**RoBERTa 40 epoch**	**DistillBERT 40 epoch**
4MLP (256-128-64-32) With 9e-5 LR , 512 batch size, 1000 epochs			

Fig. 12 The improved 4MLP's accuracy graphs on Pheme T/F dataset

pursuing these directions, we can advance the field of rumour detection and contribute to creating more effective tools for combating misinformation in the digital age.

5 Conclusion

This study makes significant contributions to the fields of rumour detection on Twitter and text embeddings. Our findings enhance our understanding of the challenges posed by Twitter data and the nuances of text representations. We provide a clear and comprehensive evaluation of different text embeddings from fine-tuned BERT-based models combined with neural network classifiers for detecting rumours on Twitter, presenting results across training, validation, and testing sets. The proposed models consistently achieved high performance on different datasets and outperformed the current SOTA models. The experimental results show that the performance of the proposed models improved significantly when trained and tested on a larger amount of data which was achieved by combining all the considered datasets.

Additionally, the experiment results show that the proposed models perform better when paired with RoBERTa embeddings than with BERT or DistillBERT. However, despite the significant difference in the number of parameters among RoBERTa, BERT, and DistillBERT, the performance differences of classifier models using these embeddings are not substantial. The results also indicate that a more complex model does not guarantee a better outcome. Many studies implied that using models with a large number of trainable parameters improves performance. However, in some cases, a simple MLP model coupled with a simple LM such as DistilBERT can perform better than a combination of more complex classifiers, such as ResNet-CNN and RoBERTa.

Therefore, it is suggested that in real-world scenarios training time, cost, and computational complexity of the models should be considered carefully rather than small improvements in results. Future studies need to examine the interaction between the dataset's characteristics, algorithm models, hyperparameter settings, and splitting of data to understand the solution more comprehensively.

Acknowledgements The authors express gratitude to the Sebelas Maret University for exceptional support as a sponsor of this research.

Author contributions All of the authors contributed equally to this work. The first draft of the article was written by Rini Anggrainingsih. All authors read, commented on all the versions of the manuscript, and approved the final manuscript.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions. Sebelas Maret University Indonesia and The University of Western Australia funded this study.

Data availability The datasets are available publicly.

Declarations

Conflict of interest We are confident to confirm that we do not have any Conflict of interest associated with this publication.

References

1. Castillo C, Mendoza M, Poblete B (2011) Information credibility on Twitter. In: Proceedings of the 20th international conference companion on World Wide Web, WWW 2011, pp 675–684. https://doi.org/10.1145/1963405.1963500
2. Ito J, Song J, Toda H, Koike Y, Oyama S (2015) Assessment of tweet credibility with lda features. In: Proceedings of the 24th international conference on World Wide Web, pp 953–958
3. Zubiaga A, Liakata M, Procter R, Wong Sak Hoi G, Tolmie P (2016) Analysing how people orient to and spread rumours in social media by looking at conversational threads. PLoS ONE. https://doi.org/10.1371/journal.pone.0150989. arXiv:1511.07487
4. Hassan NY, Haggag MH (2018) Supervised learning approach for twitter credibility detection. In: 2018 13th international conference on computer engineering and systems (ICCES), pp 196–201
5. Herzallah W, Faris H, Adwan O (2018) Feature engineering for detecting spammers on twitter: modelling and analysis. J Inf Sci 44(2):230–247
6. Ma J, Gao W, Mitra P, Kwon S, Jansen BJ, Wong K-f, Cha M (2015) Detecting rumors from microblogs with recurrent neural networks. In: Proceedings of the 25th international joint conference on artificial intelligence (IJCAI 2016), pp 3818–3824
7. Ruchansky N (2017) CSI: a hybrid deep model for fake news detection. In: Proceedings of the 2017 ACM on conference on information and knowledge management, pp 797–806
8. Alkhodair SA, Ding SHH, Fung BCM (2020) Detecting breaking news rumors of emerging topics in social media. Inf Process Manage 57(2):102018. https://doi.org/10.1016/j.ipm.2019.02.016
9. Yu F, Liu Q, Wu S, Wang L, Tan T (2017) A convolutional approach for misinformation identification. In: IJCAI international joint conference on artificial intelligence, pp 3901–3907. https://doi.org/10.24963/ijcai.2017/545
10. Xu F, Sheng VS, Wang M (2020) Knowledge-Based Systems Near real-time topic-driven rumor detection in source microblogs. Knowl-Based Syst 207:106391. https://doi.org/10.1016/j.knosys.2020.106391
11. Santhoshkumar S, Babu LD (2020) Earlier detection of rumors in online social networks using certainty-factor-based convolutional neural networks. Soc Netw Anal Min 10(1):1–17
12. Bharti M, Jindal H (2021) Automatic rumour detection model on social media. In: 2020 sixth international conference on parallel, distributed and grid computing (PDGC), pp 367–371. https://doi.org/10.1109/pdgc50313.2020.9315738
13. Wang WY (2017) “Liar, liar pants on fire”: a new benchmark dataset for fake news detection. arXiv preprint arXiv:1705.00648
14. Ajao O (2018) Fake news identification on twitter with hybrid CNN and RNN models. In: Proceedings of the 9th international conference on social media and society, pp 226–230
15. Kochkina E, Liakata M, Zubiaga A (2018) All-in-one: multi-task learning for rumour verification. arXiv preprint arXiv:1806.03713
16. Karimi H, Roy P, Saba-Sadiya S, Tang J (2018) Multi-source multi-class fake news detection. In: Proceedings of the 27th international conference on computational linguistics, pp 1546–1557
17. Das SD, Basak A, Dutta S (2021) A heuristic-driven uncertainty based ensemble framework for fake news detection in tweets and news articles. Neurocomputing
18. Huang Q, Zhou C, Wu J, Liu L, Wang B (2020) Deep spatial–temporal structure learning for rumor detection on twitter. Neural Comput Appl 1–11
19. Zhang P, Ran H, Jia C, Li X, Han X (2021) A lightweight propagation path aggregating network with neural topic model for rumor detection. Neurocomputing 458:468–477
20. Zhou K, Li B (2019) Early rumour detection. In: Proceedings of the 2019 conference of the north american chapter of the association for computational linguistics: human language technologies, vol 1, pp 1614–1623
21. Ma T, Zhou H, Tian Y, Al-Nabhan N (2021) A novel rumor detection algorithm based on entity recognition, sentence reconfiguration, and ordinary differential equation network. Neurocomputing 447:224–234
22. Yuan C, Ma Q, Zhou W, Han J, Hu S (2019) Jointly embedding the local and global relations of heterogeneous graph for rumor detection. In: 2019 IEEE international conference on data mining (ICDM). IEEE, pp 796–805
23. Wu Z, Pi D, Chen J, Xie M, Cao J (2020) Rumor detection based on propagation graph neural network with attention mechanism. Expert Syst Appl 158:113595
24. Lu Y-J, Li C-T (2020) Gcan: graph-aware co-attention networks for explainable fake news detection on social media. arXiv preprint arXiv:2004.11648
25. Wang Y, Qian S, Hu J, Fang Q, Xu C (2020) Fake news detection via knowledge-driven multimodal graph convolutional networks. In: Proceedings of the 2020 international conference on multimedia retrieval, pp 540–547
26. Koloski B, Perdih TS, RobnikŠikonja M, Pollak S, Škrlj B (2022) Knowledge graph informed fake news classification via heterogeneous representation ensembles. Neurocomputing
27. Zervopoulos A, Alvanou AG, Bezas K, Papamichail A, Maragoudakis M, Kermanidis K (2022) Deep learning for fake news detection on twitter regarding the 2019 hong kong protests. Neural Comput Appl 34(2):969–982

28. Jain DK, Kumar A, Shrivastava A (2022) Canardeep: a hybrid deep neural model with mixed fusion for rumour detection in social data streams. Neural Comput Appl 1–12
29. Kotteti CMM, Dong X, Qian L (2018) Multiple time-series data analysis for rumor detection on social media. In: 2018 IEEE international conference on big data (big data). IEEE, pp 4413–4419
30. Mikolov T, Sutskever I, Chen K, Corrado G, Dean J (2013) Distributed representations of words and phrases and their compositionality. arXiv preprint arXiv:1310.4546
31. Pennington J, Socher R, Manning CD (2014) Glove: global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1532–1543
32. Joulin A, Grave E, Bojanowski P, Mikolov T (2016) Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759
33. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. Adv Neural Inf Process Syst 30
34. Devlin J, Chang M-W, Lee K, Toutanova K (2018) Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805
35. Li W, Gao S, Zhou H, Huang Z, Zhang K, Li W (2019) The automatic text classification method based on bert and feature union. In: 2019 IEEE 25th international conference on parallel and distributed systems (ICPADS). IEEE, pp 774–777
36. Sun C, Qiu X, Xu Y, Huang X (2019) How to fine-tune bert for text classification? In: China national conference on chinese computational linguistics. Springer, pp 194–206
37. González-Carvajal S, Garrido-Merchán EC (2020) Comparing bert against traditional machine learning text classification. arXiv preprint arXiv:2005.13012
38. Rietzler A, Stabinger S, Opitz P, Engl S (2019) Adapt or get left behind: Domain adaptation through bert language model finetuning for aspect-target sentiment classification. arXiv preprint arXiv:1908.11860
39. Yu J, Jiang J (2019) Adapting bert for target-oriented multimodal sentiment classification. In: IJCAI,2019. IJCAI, pp 323–326
40. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V (2019) Roberta: a robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692
41. Nguyen DQ, Vu T, Nguyen AT (2020) Bertweet: a pre-trained language model for english tweets. arXiv preprint arXiv:2005.10200
42. Müller M, Salathé M, Kummervold PE (2020) Covid-twitter-bert: a natural language processing model to analyse covid-19 content on twitter. arXiv preprint arXiv:2005.07503
43. Sanh V, Debut L, Chaumond J, Wolf T (2019) Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108
44. Jiao X, Yin Y, Shang L, Jiang X, Chen X, Li L, Wang F, Liu Q (2019) Tinybert: distilling bert for natural language understanding. arXiv preprint arXiv:1909.10351
45. Lan Z, Chen M, Goodman S, Gimpel K, Sharma P, Soricut R (2019) Albert: a lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942
46. Sun Z, Yu H, Song X, Liu R, Yang Y, Zhou D (2020) Mobilebert: a compact task-agnostic bert for resource-limited devices. arXiv preprint arXiv:2004.02984
47. Bondielli A, Marcelloni F (2019) A survey on fake news and rumour detection techniques. Inf Sci 497:38–55. https://doi.org/10.1016/j.ins.2019.05.035
48. Meel P, Vishwakarma DK (2020) Fake news, rumor, information pollution in social media and web: a contemporary survey of state-of-the-arts, challenges and opportunities. Expert Syst Appl 153:112986
49. Pamungkas EW, Basile V, Patti V (2019) Stance classification for rumour analysis in twitter: exploiting affective information and conversation structure. arXiv preprint arXiv:1901.01911
50. Wu L, Rao Y (2020) Adaptive interaction fusion networks for fake news detection. arXiv preprint arXiv:2004.10009
51. Wu L, Rao Y, Zhao Y, Liang H, Nazir A (2020) Dtca: decision tree-based co-attention networks for explainable claim verification. arXiv preprint arXiv:2004.13455
52. Wang Z, Guo Y (2020) Rumor events detection enhanced by encoding sentimental information into time series division and word representations. Neurocomputing 397:224–243
53. Wang A, Singh A, Michael J, Hill F, Levy O, Bowman SR (2018) Glue: a multi-task benchmark and analysis platform for natural language understanding. arXiv preprint arXiv:1804.07461
54. Anggrainingsih R, Hassan GM, Datta A (2021) Bert based classification system for detecting rumours on twitter. arXiv preprint arXiv:2109.02975
55. Kaliyar RK, Goswami A, Narang P (2021) Fakebert: fake news detection in social media with a bert-based deep learning approach. Multimedia Tools Appl 80(8):11765–11788
56. Gupta P, Gandhi S, Chakravarthi BR (2021) Leveraging transfer learning techniques-bert, roberta, albert and distilbert for fake review detection. In: Forum for information retrieval evaluation, pp 75–82

57. Huang Y-H, Liu T-W, Lee S-R, Calderon Alvarado FH, Chen Y-S (2020) Conquering cross-source failure for news credibility: Learning generalizable representations beyond content embedding. In: Proceedings of The web conference 2020, pp 774–784
58. Pelrine K, Danovitch J, Rabbany R (2021) The surprising performance of simple baselines for misinformation detection. In: Proceedings of the web conference 2021, pp 3432–3441
59. Taud H, Mas JF (2018) Multilayer Perceptron (MLP) Neural networks. Geomatic Approaches for Modeling Land Change Scenarios, 451–455
60. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
61. Wolf T, Debut L, Sanh V, Chaumond J, Delangue C, Moi A, Cistac P, Rault T, Louf R, Funtowicz M, et al (2019) Huggingface's transformers: state-of-the-art natural language processing. arXiv preprint arXiv:1910.03771
62. Mosbach M, Andriushchenko M, Klakow D (2020) On the stability of fine-tuning bert: misconceptions, explanations, and strong baselines. arXiv preprint arXiv:2006.04884
63. Dong X, Victor U, Chowdhury S, Qian L (2019) Deep two-path semi-supervised learning for fake news detection. arXiv preprint arXiv:1906.05659
64. Ma J, Gao W, Wong K-F (2019) Detect rumors on twitter by promoting information campaigns with generative adversarial learning. In: The world wide web conference, pp 3049–3055
65. Al-Ahmad B, Al-Zoubi A, Abu Khurma R, Aljarah I (2021) An evolutionary fake news detection method for covid-19 pandemic information. Symmetry 13(6):1091
66. Dong X, Victor U, Chowdhury S, Qian L, View P (2019) Deep two-path semi-supervised learning for fake news detection. arXiv preprint arXiv:1906.05659arXiv:arXiv:1906.05659v1
67. Goodfellow I (2015) Deep learning, vol 1. MIT press, Cambridge

Authors and Affiliations

Rini Anggrainingsih[1,2] · **Ghulam Mubashar Hassan**[2] · **Amitava Datta**[2]

✉ Rini Anggrainingsih
rini.anggrainingsih@staff.uns.ac.id; rini.anggrainingsih@research.uwa.edu.au

Ghulam Mubashar Hassan
ghulam.hassan@uwa.edu.au

Amitava Datta
amitava.datta@uwa.edu.au

[1] Information Technology and Data Science Faculty, Sebelas Maret University, Ir Sutami 36A, 57126 Surakarta, Central Java, Indonesia

[2] Computer Science and Software Engineering, The University of Western Australia, 35 Stirling Highway, Perth, WA 6009, Australia