Applied Intelligence
https://doi.org/10.1007/s10489-022-03185-0

Long text feature extraction network with data augmentation

Changhao Tang[1] · Kun Ma[1] · Benkuan Cui[1] · Ke Ji[1] · Ajith Abraham[2]

Accepted: 4 January 2022

Abstract

The spread of COVID-19 has had a serious impact on either work or the lives of people. With the decrease in physical social contacts and the rise of anxiety on the pandemic, social media has become the primary approach for people to access information related to COVID-19. Social media is rife with rumors and fake news, causing great damage to the Society. Facing shortages, imbalance, and nosiness, the current Chinese data set related to the epidemic has not helped the detection of fake news. Besides, the accuracy of classification was also affected by the easy loss of edge characteristics in long text data. In this paper, long text feature extraction network with data augmentation (LTFE) was proposed, which improves the learning performance of the classifier by optimizing the data feature structure. In the stage of encoding, Twice-Masked Language Modeling for Fine-tuning (TMLM-F) and Data Alignment that Preserves Edge Characteristics (DA-PEC) was proposed to extract the classification features of the Chinese Dataset. Between the TMLM-F and DA-PEC processes, we use Attention to capture the dependencies between words and generate corresponding vector representations. The experimental results illustrate that this method is effective for the detection of Chinese fake news pertinent to the pandemic.

Keywords COVID-19 · Fake news · Social media · Data augmentation · Long text

1 Introduction

We-media (also called as self-media) is a platform on the Internet. It provides a user the facility to write article. To draw the attention of the public, fake news has emerged in endlessly, which not only misleads the readers who have no ideas about the truth but also leads negative influences towards our Society [1]. The virtual social networking space has become the center of cyber violations such as the promotion of false information, terrorist ideas and online rumors. Moreover, it also has become the channel and tool for malicious social manipulation by certain political and special interests groups [2].

Starting from 2019, the outbreak of novel Corona Virus has spread worldwide, leading to an ongoing pandemic [3]. Subsequently, it was declared as a global emergency by the World Health Organization (WHO) [4]. Until the beginning of April 2021, more than 134 million were confirmed and more than 2.9 million died from the epidemic worldwide. As the outbreak of COVID-19, fake news on social media has become more seriously [5]. Sina Weibo is one of the most used social media platforms in China to obtain the news of COVID-19, and it is of crucial importance to ensure the authenticity of news published in this platform [6]. The deficiency of effective supervision mechanism on a social media platform leads to many fake news and online rumors [7]. While it has exacerbated the panic of public about the COVID-19 pandemic [8], it has also constituted a probable threat to the public health [9]. Some fake news on COVID-19 could lead people to make decisions that may be harmful to the health [5]. For example, alcohol poisoning

✉ Kun Ma
ise_mak@ujn.edu.cn

Changhao Tang
201921200661@mail.ujn.edu.cn

Benkuan Cui
202021100414@mail.ujn.edu.cn

Ke Ji
ise_jik@ujn.edu.cn

Ajith Abraham
ajith.abraham@ieee.org

[1] Shandong Provincial Key Laboratory of Network Based Intelligent Computing, University of Jinan, Jinan 250022, China

[2] Machine Intelligence Research Labs, Scientific Network for Innovation and Research Excellence, Auburn, USA

Published online: 04 April 2022

has skyrocketed in Iran amid the coronavirus pandemic [10]. Therefore, we should pay more attention to fake news to avoid more serious consequences in this special period [5].

Many experts and scholars in the field of Natural Language Processing (NLP) have optimized for the problems of social media information. Correlative Denoising Autoencoder (CoDAE) uses three separated autoencoders to learn the relevance of social media information, which solves the sparse problem existing in the recommender system [11]. Sparse Stacked Denoising Autoencoder (SSDAE) solves the problem of data sparsity and imbalance in social networks by learning low-level and high-level features from social information [12]. Current research in the identification of fake news on microblogs focuses mainly on the analysis of text features, and classification or ranking methods to evaluate the credibility of microblog information [13]. The research in this field mainly has the following three problems: Firstly, microblog data contains a lot of noisy data such as emoticons, punctuation, and unreadable codes. Secondly, the Chinese dataset of the epidemic are scarce and unbalanced [6]. Finally, long texts tend to lose edge characteristics due to the cropping when performing data alignment during model training, and the retained features are difficult to ensure actual effectiveness. All these lead to severe over-fitting and under-fitting issues, which leads to poor model learning. Besides, the long text cannot be captured using any longer-term dependency beyond the predefined context length [14], which brings challenges for the feature learning task of the classifier.

Addressing all the above issues, this paper proposed a long text feature extraction network with data augmentation (LTFE). Data augmentation can improve the feature learning ability of classification model. Typical data augmentation algorithms improve the generalization capability of the model by training the network on similar but different examples [15]. Meta-learning based noise-tolerant (MLNT) training improves the generalization ability of noise fitting by generating synthetic noisy labels to simulate actual training [16]. A multi-phase blending method with incremental intensity is proposed to improve the accuracy of target detector and the generalization ability of detection network [15]. Based on data augmentation, this paper proposes a long text data processing method. By optimizing the long text part of the training set, the learning ability of the classification model for edge features is improved.

Our framework acts as a powerful and accurate feature extractor, which uses unique data augmentation methods to improve the robustness of the classifier. It includes the word masking twice of training data and the reconstruction of long text data etc. According to the characteristics of Chinese data, the network preprocesses data to reduce data noise. By optimizing the feature extraction method of long text, LTFE enhances the ability of feature extraction from the unbalanced dataset. A BERT-based (Bidirectional Encoder Representations from Transformers) encoder layer was proposed to capture long-term dependence. It uses self-attention to enhance the learning of contextual semantics and dependencies between words. However, "masked language model" (MLM) objective used by BERT during pre-training are absent from real data at fine-tuning time, resulting in the pretrain-finetune discrepancy [17]. A new learning objective is proposed in this paper, which makes the model undergo twice word masking training in the fine-tuning stage, to maintain the same learning objective of the model in the two stages. To address the issues of excessively high data dimension and missing edge features of traditional methods, this paper has proposed a data alignment method for long text. The data structure is optimized by the reconstruction of long text feature vectors. Our LTFE allows the combination of different classifiers, such as CNN and RNN, which is determined by the specific application scenarios.

Our main contributions are summarized as follows:

- **Twice-masked language modeling for fine-tuning (abbreviated as TMLM-F).** TMLM-F is a text masking learning method for fine-tuning tasks. Unlike the MLM objective of BERT, TMLM-F adds artificial symbols like [MASK] to the real data at the fine-tuning time. This method eliminates the pretrain-finetune discrepancy. To avoid key characteristics from being masked, random masking on the same data twice are carried out. It can not only increase the amount of data but also paved the way for subsequent vector reconstruction.
- **Data alignment that preserves edge characteristics (abbreviated as DA-PEC).** DA-PEC is a data reconstruction method for long texts. Compared with traditional data alignment methods, it carries on the global reconstruction to the double vectors after TMLM-F. Then, the connected vectors are cut out according to the uniform length so that the incomplete vectors are discarded. In this way, the edge characteristics of the data feature vectors that are easy to be clipped can be retained, and the accuracy of feature learning can be improved.
- **Combination novelty.** This paper combines several different concepts and methods on the basis of fake news detection to form a new combination mode. It mainly includes the following parts: COVID-19, fake news, social media, data enhancement, long text data reconstruction. This paper analyzes the characteristics of Chinese news in social media, points out the existing problems, and solves them. The data set is optimized based on data enhancement to improve the learning efficiency of the neural network model.

The remainder of the paper is organized as follows. The current work of feature extraction and fake news detection is discussed in Section 2. The problem definition related to fake news detection is discussed in Section 3. LTFE framework is illustrated in Section 4. Finally, brief conclusions and future research directions are outlined in the last section.

2 Related works

2.1 Feature extraction

The first step of natural language processing (NLP) is to convert the unstructured character such as text into a numerical representation. The result is a computer capable of "understanding" the contents of documents, including the contextual nuances of the language within them [18]. The vectorization of the text is formed by one-hot encoding in traditional NLP. Although one-hot encoding is simple, it fails to capture the rich relational structure of the lexicon and is prone to dimensional disaster [19].

Word embedding is proposed in response to the above problems and is also an important factor for improving the performance of the model [20]. Each word or phrase from the vocabulary is mapped to an N-dimension vector of real numbers by word embedding [21]. After the NNLM model [22] was proposed in 2003, a series of word vectorization technologies have also been proposed, including Word2vec, GloVe, and FastTest, which provides a variety of numerical representation methods for text. These technologies do not achieve good performance because they employ fixed vector values rather than fluid vector values for words [20]. Contextual word embedding was proposed to complement these defects, which mainly includes Embeddings from Language Models (ELMO) [23], BERT [24], and Generative Pre-Training (GPT) [25]. ELMO uses a bidirectional Long Short-Term Memory (LSTM) structure [26] and a feature-based approach. On the other hand, BERT and GPT use a transformer structure and a fine-tuning approach. Embeddings from these models can combine contextual semantics with static word vectors to generate dynamic word vectors.

The current research on long text representation methods has the problem of either serious information loss or high dimension [27]. In feature learning of long text, most deep learning models adopt clipping for data alignment [28]. A threshold of equal length is set before data preprocessing typically. The data part exceeding the threshold value is cut, while the data part less than the threshold value is padded. In this way, the long text is easy to lose the edge characteristics, and the retained part after truncation will have a lot of features that have no or little impact on classification, which affects the classification performance [28].

2.2 Fake news detection

Fake news is defined in much of the literature as news that is deliberately fabricated and provably false [7]. Many fake news detection algorithms distinguish news based on its features, which are mainly statistical or semantic features extracted from the news content [29].

The research on fake news detection is mainly divided into traditional machine learning [30] and deep learning based models [31]. Traditional machine learning mainly relies on textual hand-crafted features [32]. Typical traditional learning classifiers include Naive Bayes classifier, Support Vector Machine (SVM), Decision Tree, etc [21]. Unfortunately, the hand-crafted textual features are difficult to design [33], since the linguistic patterns are highly dependent on specific events and corresponding domain knowledge [31]. Compared with traditional machine learning, deep neural networks can learn accurate representations for textual content [34]. Recurrent Neural Network (RNN) was first applied to fake news detection of social media in 2016 [35]. It modeling the representations of posts in a time series as textual features. In 2017, Convolutional Neural Network (CNN) was applied to text classification tasks to obtain local and global features from news posts [36]. Subsequently, more fusion models are proposed by combining the advantages of neural network models like region-based CNN (RCNN) and C-LSTM. A novel end-to-end deep residual convolutional dehazing network (DRCDN) combines two different subnetworks to improve the learning ability of local and global features [37]. It captures global structure information by context aggregation subnetwork and local features by a novel hierarchical convolutional neural network. Experimental results show that deep learning based model exhibits good performance [32]. But both CNN and LSTM have natural shortcomings in the extraction of long text semantic features [38]. CNN is not capable to capture long-term dependencies due to the size limitation of the convolution kernel. Although LSTM overcomes the difficulty of the disappearance of gradients on long-distance dependence of RNN [39], it still cannot effectively capture the long-distance dependence.

Google proposed Transformer in 2017, whose Attention mechanism is widely used in various Seq2Seq models [40]. Subsequently, ELMO [23] was proposed to build a language model by bidirectional LSTM. Compared with constructing traditional context-free word vectors, it was capable to model polysemous words and get improving on six NLP tasks. However, as a serial mechanism with a long training time, LSTM is not so sufficient in feature extraction compared with Transformer. GPT was brought

up to deal with these problems [25]. It uses Transformer instead of LSTM to capture long-distance language. Both ELMO and GPT are unidirectional, which limits the choice of architectures that can be used during pre-training [24]. BERT alleviates the unidirectionality constraint by using an MLM pre-training objective [24]. However, the MLM objective was adopted by BERT only in the pre-training stage, which brings up learning biases in the pre-training and fine-tuning stages [17]. exBAKE [20] is an improved model based on the BERT pre-training. It strengthened the understanding of the semantic by alleviating data imbalance. BERT-BiLSTM-Capsule [41] is a rumor detection model based on transfer learning. It can solve the sentence-level propaganda classification problem by using a unified neural network. Same as other auto-encoding models, the BERT-based model has a learning deviation in the fine-tuning stage compared with the pre-training stage [17].

Although deep learning has an excellent performance in natural language processing, it still has some limitations. These limitations are mainly concentrated in two aspects [42]. On the one hand, these algorithms generally require a large amount of labeled data for training, and the calculation cost is high. On the other hand, DL can generate the question of underfitting and overfitting. "Transfer learning" is proposed to overcome these [43], it takes the training model for some tasks as the initial training model of the target downstream task. For example, the pre-trained 12-layer BERT model is fine-tuned to be used for the downstream task of fake news detection [42]. At the same time, it has become a research trend to combine with other neural networks based on the advantages of BERT's ability to capture sentence semantics and long-distance dependencies. Some scholars proposed a BERT-based model combining CNN which have different kernel sizes and filters [44], and it effectively addresses ambiguity.

3 Problem definition

3.1 Fake news

The broad definition of fake news focuses on the authenticity of news content [45, 46]. For example, some literature directly identifies deceptive news as fake news [30], including serious fabrications, satire, unintentional misinformation, and hoaxes. Narrowly speaking, fake news is the false articles intentionally or verifiably, which could mislead the readers [7]. This definition, which is widely used in research, includes two key features: First, it could be verified to false.Second, it is misleading with dishonest intention.

The narrow definition is adopted in this paper for the following reasons: Firstly, the intention behind fake news provides both theoretical and practical value [47], and enabling us to deeply analyze the emergence of fake news on social media as the new top epidemic. Secondly, a technology that can detect fake news under a narrow definition can be applied to a broader definition.

3.2 Fake news detection

The purpose of fake news detection is to predict whether news article A is fake news. We define the detection of fake news in the COVID-19 epidemic as a binary classification problem, because fake news is essentially a distortion bias on information manipulated by the publisher. We want to get a prediction function:

$$\mathcal{F} : \mathcal{E} \rightarrow \{0, 1\} \tag{1}$$

If a is a piece of fake news, $\mathcal{F}(a) = 1$. If a is real news, $\mathcal{F}(a) = 0$.

3.3 Architecture of fake news detection

The framework of the fake news detection system is as shown in Fig. 1. The main process includes text preprocessing, feature extraction, encoder, and decoder. Firstly, the original microblog data is processed through text preprocessing to reduce noise data. Secondly, TMLM is used to mask each long text to get two different token sequences. Thirdly, the token sequence is input into a BERT-based BERT layer to capture sentence semantics and long-distance dependencies. Fourthly, DA-PEC is used to reconstructed the two masked vectors to obtain the feature vector containing the whole text information. Finally, the prediction results are obtained by decoding through the connecting classifier. Significantly, our framework can connect different classifiers (such as CNN and RNN) according to actual application scenarios to obtain the most suitable classification model.

Formally, given $X = \{X_1, X_2, ..., X_N\}$ as the input with N long text instances, each instance X_n is processed by the TMLM-F method to produce two new instances: $X'_1 \epsilon R^r$ and $X'_2 \epsilon R^r$, where r represents the instances generated by random masking. Secondly, the BERT pre-training model is used to vectorize the token sequence and obtained two vectors which are V_1 and V_2. Then, DA-PEC is used to connect V_1 with V_2 to obtain a new one-dimensional instance V'. After the reconstruction of instance V', a set of feature vectors E' was obtained, containing all the features of long text. Finally, the classifier decodes the depth feature representation obtained by the framework to predict the classification label Y of instance X.

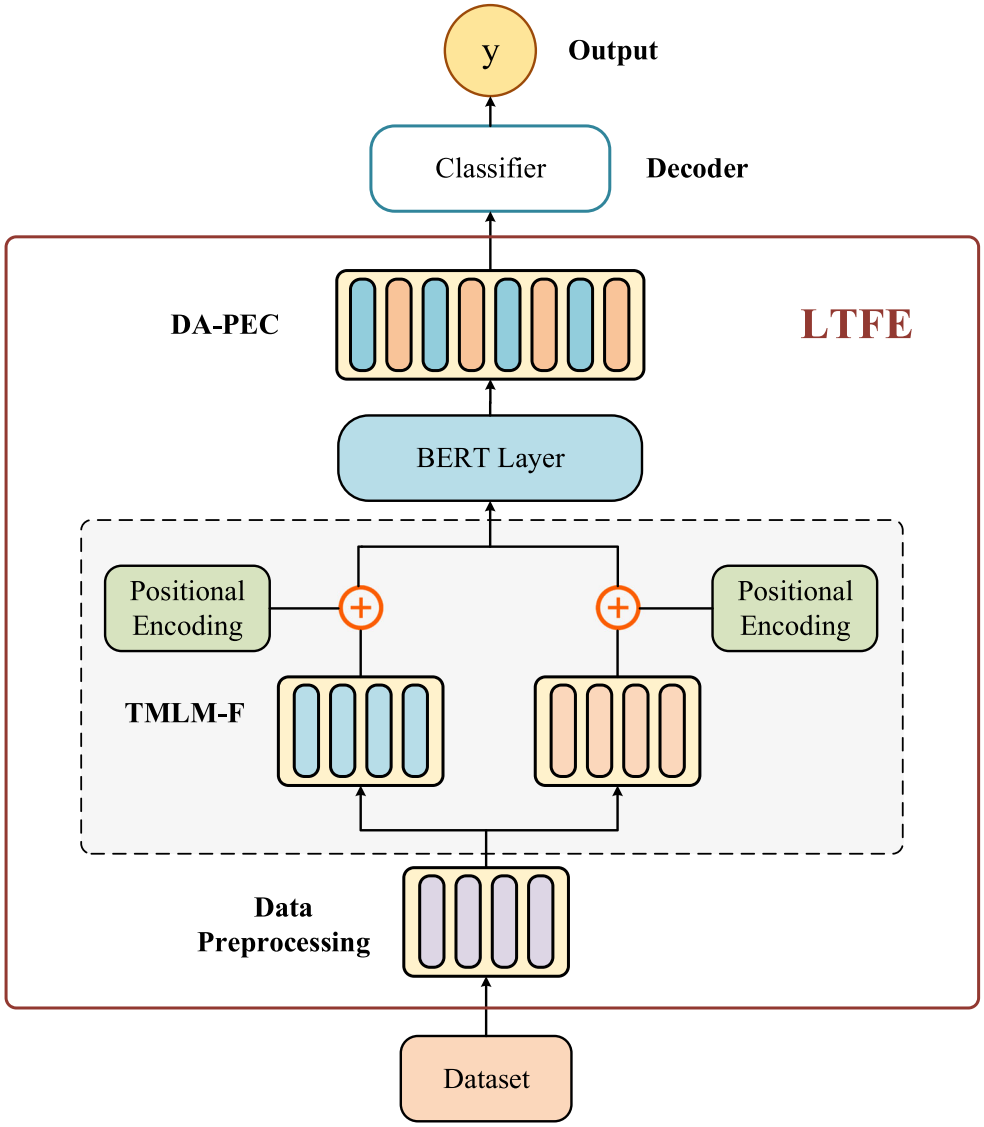

Fig. 1 General architecture of fake news detection

3.4 Metrics of fake news detection

In this section, we introduce several metrics used in the experiment to evaluate classifier performance from different perspectives.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{4}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

$Precision$(P) is used to indicate the proportion of fake news among all identifiable samples. $Recall$(R) is used to measure the proportion of all fake news samples that are correctly detected. Since the microblog data set of the COVID-19 epidemic are often asymmetric, the $Precision$ of the classifier could be improved by judging more data into the category with a larger number. Therefore, comprehensive consideration of $Precision$ and $Recall$ are required as evaluating the performance of the classifier. $F1-Score$ is the weighted harmonic average of $Precision$ and $Recall$, which can provide a more comprehensive evaluation in the detection. $Accuracy$ is used to measure the similarity between predicted fake news and the real one. It should be noted that the higher the value of the above four indicators, the better the classifier effect.

4 Methods description

4.1 Data preprocessing

The microblog data related to the new crown epidemic is not so qualified for it was obtained directly from the website. Such data is mixed with a lot of special characters, punctuation, garbled, and other data noise. It will have a negative impact on our training model without being cleaned and filtered. Therefore, before text masking, firstly we must perform data cleaning on the training dataset. Different from the English dataset, the Chinese dataset takes characters as the unit, so we need to do word segmentation to ensure the integrity of the meaning.

Firstly, the special characters and punctuation marks in the text data should be removed. We have added a large number of Weibo special symbols to the traditional stop word list since there are many emoticons on

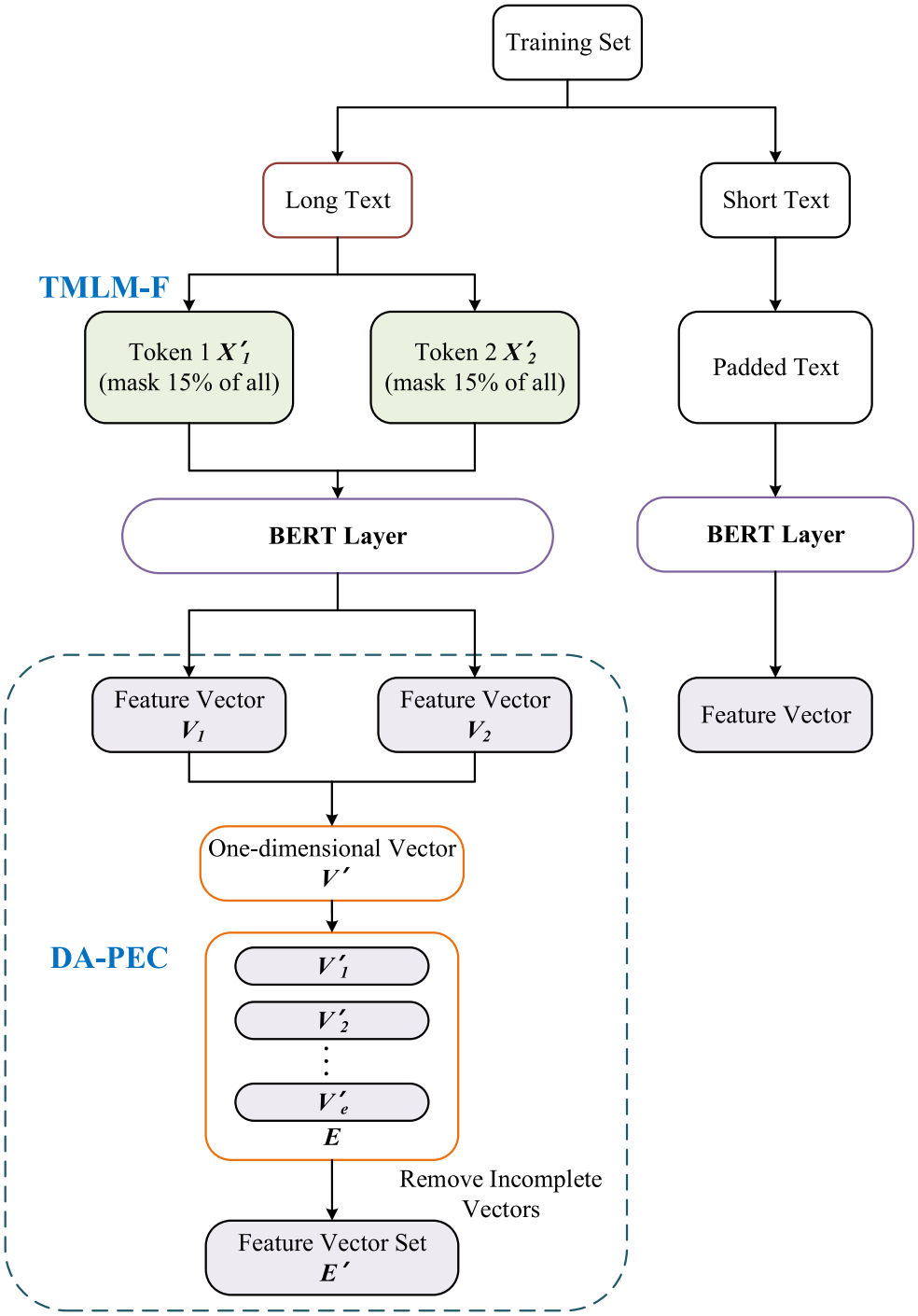

Fig. 2 Text feature extraction

Weibo. Secondly, a directed acyclic word graph (DAG) is constructed based on the Trie-tree structure, which includes all possible words in the sentence. Thirdly, dynamic programming is used to find the path of maximum probability, then the maximum segmentation combination based on word frequency is found either. Finally, for unregistered words, the Hidden Markov Model (HMM model) based on the ability of Chinese characters is used for word formation, and the Viterbi algorithm is used to obtain the hidden sequence with the greatest probability meanwhile. After the word segmentation is completed, useless words in the text information will be deleted through iterative search.

4.2 Text feature extraction

This Section mainly introduces text feature extraction shown in Fig. 2. LTFE provides a vector reconstruction method that preserves the edge characteristics of long text. Here, we divide the microblog data into two categories by length: long text and short text. Long text sequence is masked by TMLM-F twice differently. In this way, the learning bias will be reduced and the feature extraction will be more accurate when the BERT Layer is used for vectorization of the token sequence. Then, the two feature vectors from the same text are reconstructed as a whole by DA-PEC to obtain the vector set containing all the features of the long text. For short text, feature extraction is carried out in the traditional way.

Since the length of each piece of data diverse, the vector through word embedding should be processed into a uniform length first to avoid the trouble brought to the training. Firstly, a length threshold, seq_len, is set as the uniform length for data alignment. Secondly, according to the relationship between the length of each piece of data and seq_len, the data is divided into A part (less than or equal to seq_len) and B part (greater than seq_len). Thirdly, the feature vector of Part A is processed by the padding method. As for the feature vectors of data in Part B, TMLM-F and DA-PEC as described below are adopted for processing.

4.2.1 Twice-masked language modeling for fine-tuning(TMLM-F)

Word embedding is essential to improve the performance of the model. Instead of using static word embedding, contextual word embedding is adopted in this paper so that words can generate relevant vector representations within different contexts. We apply the BERT pre-training model after a lot of text training to the stage of encoding by transfer learning.

BERT uses MLM objective to masking the token sequence. The special symbol [MASK] was used to replace the certain portion of tokens which in the input sequence. Then, the model was used to recover the original tokens from the damaged version [17]. Unlike left-to-right language model pre-training, the MLM objective can not only alleviate the unidirectionality constraint, but also enable the representation to fuse the left and the right context [24].

However, BERT only used MLM objective in the pre-training stage, which would lead to the pretrain-finetune discrepancy [17]. Therefore, we introduced the Twice-Masked Language Modeling for Fine-tuning(TMLM-F), as shown in Fig. 3.

First, according to the pre-processing method mentioned in Section 4.1, the training data set is processed to obtain the Chinese word segmentation after removing redundant data. Secondly, in order to prevent key feature words from being masked, MASK mechanism was used twice for each data. 15% of BPE tokens of the data are randomly selected for masked language modeling. Finally, the text vectorization is done using BERT Tokenizer. Adding MASK mechanism in the downstream task, the learning deviations of the model in the pre-training and fine-tuning stages can be eliminated, which brings more accurate feature vectors.

Formally, given a target sequence of length n, $\boldsymbol{X} = (x_1, x_2, x_3, \ldots, x_n)$, where x represents a single word or token. We need to generate the noisy version of $\boldsymbol{X}$ after masked learning, which is denoted as $\boldsymbol{X'}$. The learning goal of TMLM-F is to predict the masked token x_m in X' based on the source sequence $\boldsymbol{X}$and the unmasked token

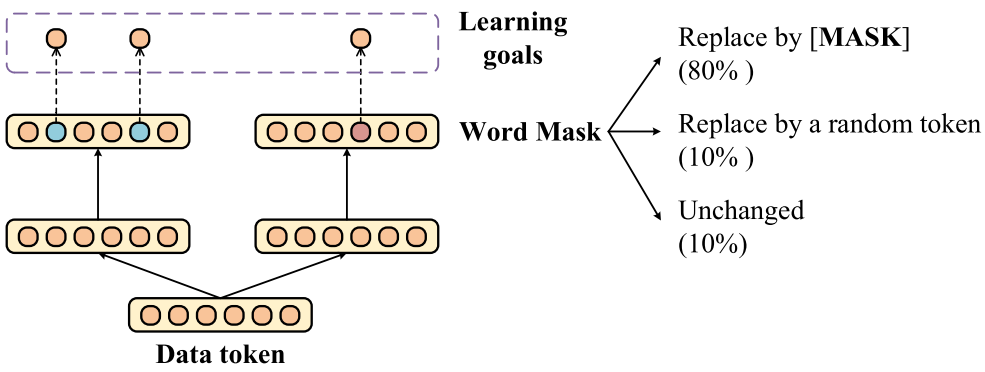

Fig. 3 Twice-Masked Language Modeling for Fine-tuning(TMLM-F)

x. In this process we set up an indicator s_i to indicate the position to be masked. When $s_i = 0$, it representing the i-th position in sequence $\boldsymbol{X}$ is not masked. When $s_i = 1$, it representing the i-th position in sequence $\boldsymbol{X}$ is masked. In the TMLM-F method, we mask the target sequence $\boldsymbol{X}$ and get the sequences $\boldsymbol{X_1'}$ and $\boldsymbol{X_2'}$. In each training, we use the cross entropy function to calculate the loss of each iteration. Since sequence X_1' and sequence X_2' are two relatively independent sequences, the cross-entropy can be calculated in parallel. Based on the MLM objective, we define the learning objective of TMLM-F as the following formula:

$$H(\theta; \boldsymbol{X}, \boldsymbol{X'}) = -\sum_{x, x' \in \boldsymbol{X}, \boldsymbol{X'}} \log p_\theta(\bar{x} \mid x') \tag{6}$$

$$\max_\theta \log p_\theta(\bar{x} \mid x') \approx \sum_{i=1}^{n} s_i \log p_\theta\left(w_i \mid x'\right) \tag{7}$$

$$p_\theta\left(w_i \mid x'\right) = \frac{\exp\left(Q_\theta(x')_i e\left(w_i\right)\right)}{\sum_{w'} \exp\left(Q_\theta(x')_i e\left(w'\right)\right)} \tag{8}$$

Where θ represents the parameters of the language model, $\bar{x}$ is the original token of each masked token, $Q_\theta(x')$ is the contextual representation of x', and $e(w_i)$ is the word embedding from the last prediction layer.

To further eliminate the learning bias at different stages, we did not replace all the selected tokens with [MASK]. When a token is selected, it has an 80% chance of being replaced with a [MASK], a 10% chance of being replaced with a random token, and a 10% chance of remaining the same. During model training, the replaced tokens are used to calculate cross entropy loss to predict the original tokens.

4.2.2 BERT layer

Some words like COVID-19, Epidemic and Test, appear frequently in the microblog data but have nothing effect to distinguish news. Each character in the microblog data has different contributions to semantic information and different effects on the detection of fake news. Therefore, the essential part of our research lies in how to master the function of each character correctly and how to dig out the original semantic information.

Motivated by the above-mentioned observations, the BERT Layer based BERT Encoder is introduced into the framework LTFE to learn the word dependencies within a sentence. The token sequence processed by TMLM-F will be input to the BERT Layer to learn attention information. Since the [MASK] label is added to the token sequence, the learning biases between pre-training and fine-tuning stages will be corrected in the training process. Because the two token sequences use different masking methods, different words will be paid different attention in model training, which makes it easier to determine key features in multiple fine-tuning stages. After attention learning, the data vector will be processed by DA-PEC method. Finally, the vector of [n,128,768] dimension is output to the classifier for decoding.

The BERT Layer is composed of a stack of 8 identical layers with four sub-layers each, including multi-head attention layer, residual connection layer, feed-forward network layer and another residual connection layer. As is shown in Fig. 1. The multi-head attention layer mainly uses the self-attention mechanism to calculate the attention of each word to other words in the text sequence, which can integrate the original semantic information into the output vector of the embedding layer.

Query, Key and Value are the three elements of self-attention, whose vector representation are all come from the same input text. Since the dataset is Chinese text data, we calculate self-attention by word. Each word in the data has its own original Value. With target word as the Query and other word in the context as the Key, the self-attention calculates the similarity between them and integrate the obtained weight into the original Value of the target word. The matrix of self-attention is as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{9}$$

4.2.3 Data alignment that preserves edge characteristics(DA-PEC)

Padding is usually used to align data in traditional model training. In this process, the text parts beyond the threshold value will be clipped, so long text is prone to lose edge features. The DA-PEC method proposed in this paper generates two different vectors from the same text and connects them to obtain a one-dimensional composite vector. Where the input of DA-PEC is the masked vectors output by TMLM-F. Then the whole vector is clipped to get the reconstructed vector set.Finally, the incomplete vectors which could easily affect the semantic integrity are eliminated and the feature vectors containing all text information are obtained. DA-PEC can be combined with TMLM and BERT Layer mentioned above to achieve better results. As a clipping strategy, DA-PEC aims to maintain the integrity of data features. It can output a vector set containing all the information of the original data, which can be used to help the neural network learn more features. DA-PEC can be regarded as composed of two main parts, as is shown in Fig. 2. It can be expressed mathematically as follows:

$$V' = \text{Con}(V_1, V_2) \tag{10}$$

$$E' = \text{Re}(\mathcal{F}(V', seq\_len)) \tag{11}$$

Where V_1 and V_1 are the two masked vectors output by TMLM-F method, Con is vector concatenating operation,

seq_len is the length of data alignment set, $\mathcal{F}$ is the overall data reconstruction method, and Re is the data screening method. Through supervised learning of vector set E', all feature information can be extracted from the original data. The specific process is as follows:

The reconstruction method for the data vector of long text is shown in Fig. 2. Firstly, add starting bits to the masked text sequence $X_1 = \left(x_1^1, x_2^1, x_3^1, \ldots, x_d^1\right)$ and $X_2 = \left(x_1^2, x_2^2, x_3^2, \ldots, x_d^2\right)$ to obtain $X_1' = \left(x_0^1, x_1^1, x_2^1, x_3^1, \ldots, x_d^1\right)$ and $X_2' = \left(x_0^2, x_1^2, x_2^2, x_3^2, \ldots, x_d^2\right)$. Here we borrow [CLS] from the BERT model to indicate the beginning of the data and the distinction from the previous data. Secondly, the BERT pre-training model is used as the embedding layer to transform the token sequence X_1' and X_2' into vector representation, and two long text feature vectors are obtained, which is $V_1 = \left(v_0^1, v_1^1, v_2^1, v_3^1, \ldots, v_d^1\right)$ and $V_2 = \left(v_0^2, v_1^2, v_2^2, v_3^2, \ldots, v_d^2\right)$. Here the BERT pre-training model can be selected according to specific application scenarios. In this paper, the RoBERTa Chinese pre-training model is selected to construct BERT Layer. Thirdly, the feature vectors $V1$ and $V2$ are connected into a one-dimensional vector $V' = \left(v_0^1, v_1^1, v_2^1, v_3^1, \ldots, v_d^1, v_0^2, v_1^2, v_2^2, v_3^2, \ldots, v_d^2\right)$. At the same time, the corresponding position vector V_{pos} is set to indicate the effective position of the vector, so that the semantic integrity of the clipped vector can still be maintained. Fourthly, a new set of feature vectors $E = \left\{V_1', V_2', \ldots, V_e'\right\}$, which includes all data features of long text, is obtained by truncating from front to back with the unit of seq_len. Finally, the part of data shorter than seq_len is removed to avoid vector sparsity and semantic ambiguity.

The above method has the following advantages: Firstly, it can solve the problem of easy losing edge characteristics in the long text by the traditional method. Secondly, it eliminate the semantic deviation caused by adding [0] to the incomplete token in the padding method. Thirdly, it can learn all the features in the short text.

5 Dataset overview

This Section mainly introduces the relevant information of the two datasets used in the experiments, which can be available at https://github.com/SmallZzz/FakeNewsData.

5.1 COVID-19 dataset

As reported in the third-quarter earnings of the company in 2020, Weibo has 511 million monthly active users and 224 million daily ones. According to a large number of active users, it is believed that Weibo is one of the most popular social media in China for people to get COVID-19 information [6].

During the training and testing of the model, CHECKED, as the first Chinese COVID-19 social media dataset, is the primary dataset used in our research. From December 2019 to August 2020, CHECKED has 2,120 pieces of blogs including 344 being “Fake” and 1776 being “Real”. The data has high quality after being strictly reviewed in

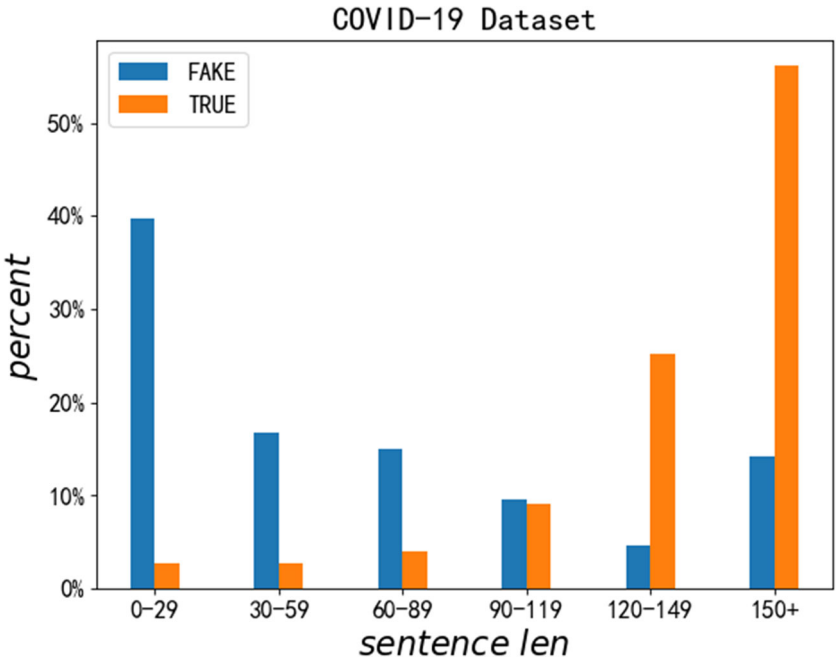

Fig. 4 COVID-19 Data length distribution

authenticity and certified by the official community management. Each microblog is represented by the following components: id, label, date, user_id, user_name, text, pic_url, video_url, comment_num, repost_num, like_num, comments and reposts. Another 718 pieces of data were added to prevent overfitting caused by the extreme shortage of dataset. Those part of data is also qualified for obtaining from Tencent fact check platform. A total of 2,838 pieces of data were combined, including 883 pieces of fake news and 1,955 pieces of true news. The data length distribution is shown in Fig. 4.

5.2 BAAI dataset

We chose a dataset from an Internet fake news detection contest for our experiment. The competition was jointly held by the Beijing Academy of Artificial Intelligence (BAAI) and the Institute of Computing Technology of the Chinese Academy of Sciences in 2019. BAAI has set up the Data Open Research Center with the aim of achieving the sharing of high-quality data and knowledge. The dataset for this competition was provided by BAAI and is of high quality. The dataset contains 38,471 items of data, including 19,186 true news and 19,285 fake news. Each piece of data is represented by the following components: id, text, and label, where 1 represents the positive example (true), 0 represents the negative example (fake). The data length distribution is shown in Fig. 5. Different from the dataset described in Section 5.1, the length of news in this dataset mostly concentrates within the range of 120-150.

6 Experimental analysis

This paper takes fake news detection as a binary-class classification task, Fake News is treated as the positive samples and Real News is treated as the negative samples. We evaluate the results with Precision (P), Recall (R), F1, and Accuracy. The method of three-fold cross-validation was adopted in the training of this paper due to the Chinese COVID-19 dataset being in shortage. We tested our approach using two different datasets, each of which was split into train set and test set according to the ratio of 4:1.

In the experiment, we fixed several hyperparameters of LTFE, such as data alignment length $seq\_len = 128$, word vector dimension $embedding\_size = 768$, and the number of attention-head in BERT Layer is $head\_num = 8$. During model training, set $num\_epochs = 10$, $batch\_size = 32$, $learning\_rate = 5e - 5$. For the model related to CNN, we also fixed the size and number of convolution kernels: $filter\_sizes = (2, 3, 4)$, $num\_filters = 256$.

6.1 Experimental results of expanded CHECKED dataset

We first choose the expanded CHECKED dataset for experiment, and the results are as shown in Table 1. TextCNN and TextRNN were selected as the baselines for testing. The TMLM-F approach cannot be used in these two models for the absence of MLM objectives. Through experiments, the result with or without DA-PEC method was obtained respectively. It's clear that although

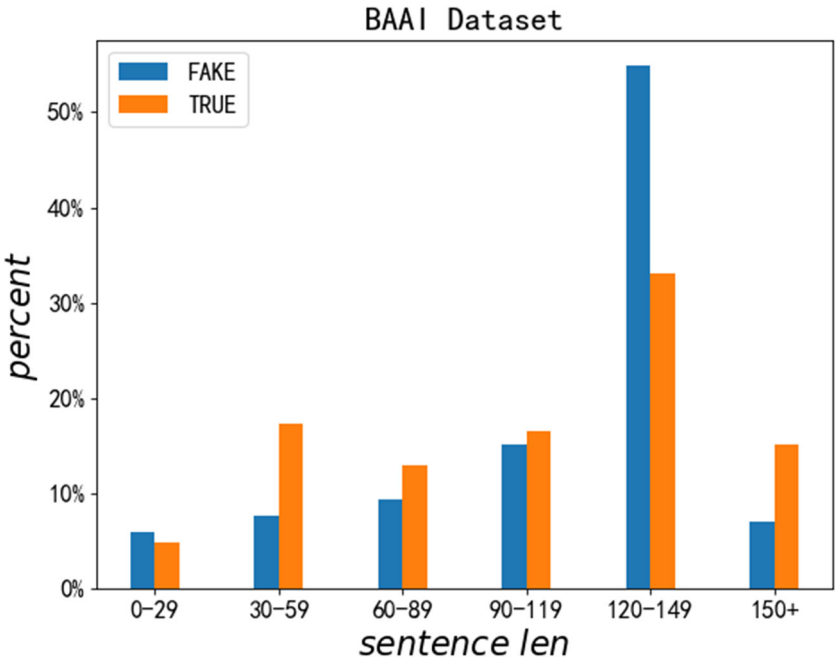

Fig. 5 BAAI data length distribution

Table 1 The results of expanded CHECKED dataset

Model	Method		P			R			F1			ACC
	TMLM-F	DA-PEC	True	Fake	Macro	True	Fake	Macro	True	Fake	Macro	
TextCNN	-	no	0.9275	0.8758	0.9017	0.9529	0.8171	0.8850	0.9400	0.8454	0.8927	0.9136
	-	yes	0.8829	0.9106	0.8967	0.9727	0.6829	0.8278	0.9256	0.7805	0.8531	0.8889
TextRNN	-	no	0.8228	0.8602	0.8415	0.9677	0.4878	0.7278	0.8894	0.6226	0.7560	0.8289
	-	yes	0.8298	0.8660	0.8479	0.9677	0.5122	0.7400	0.8935	0.6437	0.7686	0.8360
RCNN	-	no	0.8197	0.8667	0.8432	0.9702	0.4756	0.7229	0.8886	0.6142	0.7514	0.8272
	-	yes	0.8319	0.8763	0.8541	0.9702	0.5183	0.7443	0.8958	0.6513	0.7736	0.8395
DPCNN	-	no	0.9100	0.8690	0.8895	0.9529	0.7683	0.8606	0.9309	0.8155	0.8732	0.8995
	-	yes	0.9277	0.8816	0.9046	0.9553	0.8171	0.8862	0.9413	0.8481	0.8947	0.9153
BERT	no	no	0.9400	0.9267	0.9334	0.9727	0.8476	0.9101	0.9561	0.8854	0.9207	0.9365
+CNN	no	yes	0.9897	0.8939	0.9418	0.9529	0.9756	0.9642	0.9709	0.9709	0.9519	0.9594
LTFE	yes	yes	0.9872	0.9086	**0.9479**	0.9603	0.9695	**0.9649**	0.9736	0.9381	**0.9558**	**0.9630**
+CNN												
BERT	no	no	0.8902	0.8923	0.8912	0.9653	0.7073	0.8363	0.9262	0.7891	0.8577	0.8907
+RNN	no	yes	0.9736	0.8191	0.8964	0.9156	0.9390	0.9273	0.9437	0.8750	0.9094	0.9224
LTFE	yes	yes	0.9769	0.8708	**0.9238**	0.9429	0.9451	**0.9440**	0.9596	0.9064	**0.9330**	**0.9436**
+RNN												
BERT	no	no	0.8284	0.8737	0.8510	0.9702	0.5061	0.7382	0.8937	0.6409	0.7673	0.8360
+RCNN	no	yes	0.8957	0.9365	**0.9161**	0.9801	0.7195	0.8498	0.9360	0.8138	0.8749	0.9048
LTFE	yes	yes	0.9262	0.9048	0.9155	0.9653	0.8110	**0.8881**	0.9453	0.8553	**0.9003**	**0.9206**
+RCNN												
BERT	no	no	0.9317	0.8662	0.8990	0.9479	0.8293	0.8886	0.9397	0.8474	0.8935	0.9136
+DPCNN	no	yes	0.9353	0.9133	0.9243	0.9677	0.8354	0.9016	0.9512	0.8726	0.9119	0.9295
LTFE	yes	yes	0.9722	0.8947	**0.9335**	0.9553	0.9329	**0.9441**	0.9637	0.9134	**0.9386**	**0.9489**
+DPCNN												

Bold entries are the optimal results of three different experiments

the TextRNN has a high Precision, the Recall is very low. Among them, the recall rate of TextRNN for fake news samples (positive samples) is even lower than 50% when DA-PEC is not used. TextCNN also had a lower Recall for fake news after using DA-PEC. Due to the imbalanced proportion of true and fake samples in the training set, TextRNN, with more errors in the learning of classification features, has poor learning ability for the features of fake news. By comparing the two groups of experiments, it is obvious that the performance of TextCNN is better overall.

We further tested the effect of RCNN and DPCNN on this dataset. RCNN is a model combining RNN with CNN, which incorporates contextual semantic information in the word embedding. DPCNN is an improved model of CNN, which strengthens the learning of long-term dependence by deepening the network. From the experimental results of the above four models, it is clear that TextRNN, RCNN, and DPCNN get better results after using DA-PEC to process long text instead of the traditional padding method. DPCNN has the highest Accuracy, which is 0.9153. In addition, we can see that the Precision and Recall of both positive and negative samples are improved, which indicates that our method has high accuracy in learning different sample features. Traditional data alignment methods crop the data according to a preset uniform length, and those exceeding expectations are discarded. The key features that originally existed in the edge part were also discarded, making the model unable to learn this effective information in the training process. DA-PEC generates a vector set with all the information of the original data through overall reconstruction of the long text vector. In the process of model learning, this effective information can be captured in different degrees, thus improving the accuracy of classification. Combined with the following experimental results, it can be proved that DA-PEC shows great superiority in processing long text. The overall reconstruction of composite vectors from the same text can effectively preserve the edge features of long text and improve the efficiency of model learning.

In the following experiment, the BERT Layer was used to perform model fusion with CNN, RNN, RCNN and DPCNN respectively. Different classifiers were used to decode the feature vectors encoded by BERT Layer, and the classification results were obtained. We respectively tested

the experimental effects of the two models with and without DA-PEC and reached three conclusions. First, experimental results show that the model encoded by BERT Layer is better than the original one under the same method. BERT enables the representation to fuse the left and the right context by using MLM objective, we believe that BERT can better capture long-term dependencies and enhance the feature learning of the model. Second, the accuracy of all models using the DA-PEC method is improved, among which Bert +CNN is increased by 2.29%, Bert +RNN is increased by 3.17%, Bert +DPCNN is increased by 1.56%, and Bert +RCNN has the largest improvement, which is 6.88%. Third, the Recall of all the above models was significantly improved after the adoption of DA-PEC, especially for the fake sample. According to Fig. 4, we can see that the CHECKED data is very unbalanced, which poses a challenge to the accuracy of feature extraction. The Attention mechanism learns information from different subspaces through Scaled Dot-product Attention (SDA), making the key information in the decoding vector easier to learn. The complete information retained by DA-PEC method is encoded by BERT Layer and output to feature vector. The experimental results show that the effective information in these feature vectors is more than that in traditional vectors. Through the above experiments, we further prove that DA-PEC can effectively maintain the edge features of long text.

The experimental effects of the LTFE framework combined with CNN, RNN, RCNN and DPCNN were tested respectively in the following experiments. LTFE combines TMLM-F, BERT Layer and DA-PEC. According to its great advantages in semantic learning, BERT Layer was used in this framework to extract text features, and TMLM-F method will further eliminate its learning bias in the pre-training and fine-tuning stages as well. Finally, DA-PEC is used to reconstruct the feature vectors obtained by the above method to obtain comprehensive and accurate feature information. The experimental results show that using our framework is more effective than using BERT alone. LTFE framework introduces the TMLM-F method, which can achieve the highest accuracy compared with the combined method of BERT Layer and DA-PEC. It's worth noting that LTFE+CNN obtained the highest Accuracy of all experiments, which was 0.9630. At the same time, most of the classifiers using LTFE have the highest macro-averaging of P, R and F1. Although the addition of BERT makes the understanding of the model more accurate for semantics, the lack of MLM objective in the fine-tuning stage leads to errors in the learning of the model. Our framework uses the TMLM-F method to introduce the MLM objective into the fine-tuning stage. By corresponds to the learning objectives in the pre-training stage of the Attention model, the two-stage learning bias is eliminated, and thus more excellent results can be achieved. BERT Layer can enhance the extraction ability of reserved features in DA-PEC, and TMLM-F can also eliminate the learning bias of BERT Layer. Through the combination of three different methods, effectively make up for each other's shortcomings. Therefore, the LTFE framework is able to demonstrate optimal encoding capabilities.

The following analysis was conducted given the above experimental results: Firstly, the DA-PEC algorithm can effectively deal with the easily losing edge characteristics in long text data so that the model can learn all text features. Secondly, the TMLM-F algorithm is used to solve the learning bias of the pre-training model in the pre-training and fine-tuning stages, which enhanced the model in learning specific semantics. Finally, the LTFE framework proposed in this paper makes up for the defects of some classification models, such as CNN, in capturing long-term dependence by introducing the self-attention mechanism.

6.2 Experimental results of BAAI dataset

Similar to the experimental process in Section 6.1, we take TextCNN, TextRNN, RCNN, and DPCNN as the basic models. The experimental results are as shown in Table 2. The data set used in this experiment is BAAI data. As can be seen from Fig. 5, it is also an unbalanced data set. However, this data set has no obvious length distribution characteristics, and both true news and fake news are mostly distributed within the interval of 120-149. This distribution reduces the error caused by learning the length characteristics of the model, but has a high requirement on the accuracy of feature learning. We conducted a number of experiments to test the positive effects of the combination of these three approaches. We tested the effects of these four models without and with the DA-PEC method. The four models were all improved by the DA-PEC method after testing, with an average increase of 1.0525%. After that, we combined the BERT Layer as the encoder with TextCNN, TextRNN, RCNN, and DPCNN respectively to obtain a set of experimental results. It is clear that compared with the model without BERT Layer, adding the self-attention mechanism in encoding can greatly improve the learning effect of the model. TextCNN cannot learn the long-distance dependence relationship due to the limitation of the convolution kernel size. RNN also has the difficulty of the disappearance of gradients. When the data is long, the BERT layer has a more obvious advantage in learning long-distance dependencies. On this basis, we further test the effect of the fusion model of BERT Layer and other models in the following two cases: using the TMLM-F method only; using TMLM-F and DA-PEC methods (that is, using LTFE framework). The results show that, firstly, the performance of the model is improved by adding our method in most

Table 2 The results of BAAI dataset

Model	Method		P			R			F1			ACC
	TMLM-F	DA-PEC	True	Fake	Macro	True	Fake	Macro	True	Fake	Macro	
TextCNN	-	no	0.8466	0.9400	0.8933	0.9447	0.8349	0.8898	0.8929	0.8844	0.8886	0.8888
	-	yes	0.8644	0.9442	0.9043	0.9475	0.8568	0.9021	0.9041	0.8984	0.9012	0.9013
TextRNN	-	no	0.8306	0.9278	0.8792	0.9341	0.8164	0.8752	0.8793	0.8685	0.8739	0.8741
	-	yes	0.8623	0.9125	0.8874	0.9145	0.8592	0.8869	0.8876	0.8850	0.8863	0.8863
RCNN	-	no	0.8638	0.8886	0.8762	0.8874	0.8651	0.8763	0.8754	0.8767	0.8761	0.8761
	-	yes	0.8634	0.9080	0.8857	0.9095	0.8613	0.8854	0.8859	0.8841	0.8850	0.8850
DPCNN	-	no	0.8919	0.9218	0.9068	0.9215	0.8923	0.9069	0.9064	0.9068	0.9066	0.9066
	-	yes	0.8753	0.9621	0.9187	0.9645	0.8675	0.9160	0.9177	0.9124	0.9151	0.9151
BERT	no	no	0.9496	0.9730	0.9613	0.9726	0.9502	0.9614	0.9609	0.9614	0.9612	0.9612
+CNN	no	yes	0.9566	0.9674	0.9620	0.9665	0.9577	0.9621	0.9615	0.9625	0.9620	0.9620
LTFE	yes	yes	0.9560	0.9772	**0.9666**	0.9768	0.9567	**0.9667**	0.9663	0.9668	**0.9665**	**0.9665**
+CNN												
BERT	no	no	0.9600	0.9597	0.9598	0.9587	0.9609	0.9598	0.9594	0.9603	0.9598	0.9598
+RNN	no	yes	0.9677	0.9627	**0.9652**	0.9616	0.9686	**0.9651**	0.9647	0.9657	**0.9652**	**0.9652**
LTFE	yes	yes	0.9594	0.9691	0.9643	0.9687	0.9599	0.9643	0.9640	0.9645	0.9643	0.9643
+RNN												
BERT	no	no	0.9586	0.9659	0.9623	0.9648	0.9599	0.9623	0.9617	0.9629	0.9623	0.9623
+RCNN	no	yes	0.9528	0.9707	0.9617	0.9701	0.9537	0.9619	0.9614	0.9621	0.9617	0.9617
LTFE	yes	yes	0.9647	0.9632	**0.9640**	0.9617	0.9661	**0.9639**	0.9632	0.9646	**0.9639**	**0.9639**
+RCNN												
BERT	no	no	0.9726	0.9454	0.9590	0.9416	0.9744	0.9580	0.9569	0.9597	0.9583	0.9583
+DPCNN	no	yes	0.9556	0.9631	0.9594	0.9620	0.9569	0.9595	0.9588	0.9600	0.9594	0.9594
LTFE	yes	yes	0.9619	0.9683	**0.9651**	0.9673	0.9631	**0.9652**	0.9646	0.9657	**0.9652**	**0.9652**
+DPCNN												

Bold entries are the optimal results of three different experiments

cases. Secondly, models using the LTFE framework can achieve the best results under the same conditions. It is worth noting that the accuracy of LTFE+RNN is slightly reduced due to the use of DA-PEC method, but it is still higher than that of BERT+RNN. And the accuracy of Bert Layer+RNN is reduced by 0.06% after using TMLM-F. We think this is related to the basic model itself and the dataset, and within a reasonable margin of error. DA-PEC preserves all the information of long text, which facilitates most models. However, RNN has a learning error in the training process due to its learning disadvantage of long-distance dependence. TMLM-F introduces MLM objective to the fine-tuning stage, which can improve the negative problems caused by BERT Layer. For most models, the combination of TMLM-F and BERT Layer can bring great improvement. Compared with the experimental results of the four basic models, the accuracy of the model improved by 7.8575% on average after using the LTFE framework. Among them, LTFE+CNN has the largest improvement, which is 96.65%.

7 Conclusions

In this paper, we analyzed the problems faced by the Chinese epidemic fake news detection and propose a neural network framework to solve it. Unlike most methods that use clipping for data alignment, we propose a data reconstruction method for long text. This method can effectively preserve the edge characteristics of long text and make the classifier learn the classification features more perfectly. To get a more accurate representation of features, we incorporate contextual information through a Bert-based attention mechanism to capture the important components in post. At the same time, to eliminate a pretrain-finetune discrepancy caused by MLM objective, we propose Twice-Masked Language Modeling for Fine-tuning. The proposed LTFE allows the combination of the different classifiers to handle different situations. The experimental results illustrate that the proposed method can effectively and stably detect Chinese social media fake news.

Acknowledgements This work was supported by the National Natural Science Foundation of China (61772231), the Shandong Provincial Natural Science Foundation (ZR2017MF025), the Project of Shandong Provincial Social Science Program (18CHLJ39), the Project of Independent Cultivated Innovation Team of Jinan City (2018GXRC002), and the Shandong Provincial Key R&D Program of China (2021CXGC010103).

Availability of Data and Material The data can be available at https://github.com/SmallZzz/FakeNewsData

Code Availability The code are available from the corresponding author on reasonable request.

Declarations

Conflict of Interests The authors declare that they have no conflict of interest.

References

1. McGonagle T (2017) "fake news" false fears or real concerns? Netherlands Quarterly of Human Rights 35(4):203–209
2. McPherson M, Smith-Lovin L, Cook JM (2001) Birds of a feather: Homophily in social networks. Annual review of sociology 27(1):415–444
3. Sohrabi C, Alsafi Z, O'Neill N, Khan M, Kerwan A, Al-Jabir A, Iosifidis C, Agha R (2020) World health organization declares global emergency: A review of the 2019 novel coronavirus (covid-19). International journal of surgery 76:71–76
4. Nicola M, Alsafi Z, Sohrabi C, Kerwan A, Al-Jabir A, Iosifidis C, Agha M, Agha R (2020) The socio-economic implications of the coronavirus and covid-19 pandemic: a review. International journal of surgery
5. Patwa P, Sharma S, PYKL S, Guptha V, Kumari G, Akhtar MS, Ekbal A, Das A, Chakraborty T (2020) Fighting an infodemic: Covid-19 fake news dataset. arXiv:2011.03327
6. Yang C, Zhou X, Zafarani R (2020) Checked: Chinese covid-19 fake news dataset. arXiv:2010.09029
7. Allcott H, Gentzkow M (2017) Social media and fake news in the 2016 election. Journal of economic perspectives 31(2):211–36
8. Huynh TL et al (2020) The covid-19 risk perception: A survey on socioeconomics and media attention. Econ. Bull 40(1):758–764
9. Lampos V, Moura S, Yom-Tov E, Cox IJ, McKendry R, Edelstein M (2020) Tracking covid-19 using online search. arXiv:2003.08086
10. Rostami M (2020) The coronavirus disease 2019 (covid-19) and alcohol use disorders in iran. American Journal of Men's Health 14(4):1557988320938610
11. Pan Y, He F, Yu H (2020) A correlative denoising autoencoder to model social influence for top-n recommender system. Frontiers of Computer science 14(3):1–13
12. Pan Y, He F, Yu H (2020) Learning social representations with deep autoencoder for recommender system. World Wide Web 23(4):2259–2279
13. Kumar S, Shah N (2018) False information on web and social media: A survey. arXiv:1804.08559
14. Dai Z, Yang Z, Yang Y, Carbonell J, Le QV, Salakhutdinov R (2019) Transformer-xl: Attentive language models beyond a fixed-length context. arXiv:1901.02860
15. Quan Q, He F, Li H (2021) A multi-phase blending method with incremental intensity for training detection networks. Vis Comput 37(2):245–259
16. Li J, Wong Y, Zhao Q, Kankanhalli MS (2019) Learning to learn from noisy labeled data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5051–5059
17. Yang Z, Dai Z, Yang Y, Carbonell J, Salakhutdinov R, Le QV (2019) Xlnet: Generalized autoregressive pretraining for language understanding. arXiv:1906.08237
18. Rakhlin A (2016) Convolutional neural networks for sentence classification. GitHub
19. Zhang W, Zhang Y, Yang K (2019) Optimizing word embedding for fine-grained sentiment analysis. In: International Conference on Artificial Intelligence and Security, Springer, pp 275–286
20. Jwa H, Oh D, Park K, Kang JM, Lim H (2019) exbake: automatic fake news detection model based on bidirectional encoder representations from transformers (bert). Appl Sci 9(19):4062
21. Kowsari K, Jafari Meimandi K, Heidarysafa M, Mendu S, Barnes L, Brown D (2019) Text classification algorithms: A survey. Information 10(4):150
22. Bengio Y, Ducharme R, Vincent P, Janvin C (2003) A neural probabilistic language model. The journal of machine learning research 3:1137–1155
23. Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L (2018) Deep contextualized word representations. arXiv:1802.05365
24. Devlin J, Chang M-W, Lee K, Toutanova K (2018) Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805
25. Radford A, Narasimhan K, Salimans T, Sutskever I (2018) Improving language understanding by generative pre-training
26. Schuster M, Paliwal KK (1997) Bidirectional recurrent neural networks. IEEE transactions on Signal Processing 45(11):2673–2681
27. Yang C (2018) Research on fast and precise classification algorithm of long text based on fasttext
28. Wenshuai Q (2020) Research on long news texts representation and classification method based on network model fusion
29. Gupta A, Kumaraguru P, Castillo C, Meier P (2014) Tweetcred: Real-time credibility assessment of content on twitter. In: International Conference on Social Informatics, Springer, pp 228–243
30. Rubin VL, Chen Y, Conroy NK (2015) Deception detection for news: three types of fakes. Proceedings of the Association for Information Science and Technology 52(1):1–4
31. Ruchansky N, Seo S, Liu Y (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
32. Wang Y, Ma F, Jin Z, Yuan Y, Xun G, Jha K, Su L, Gao J (2018) Eann: Event adversarial neural networks for multi-modal fake news detection. In: Proceedings of the 24th acm sigkdd international conference on knowledge discovery & data mining, pp 849–857
33. Wang Y, Yang W, Ma F, Xu J, Zhong B, Deng Q, Gao J (2020) Weak supervision for fake news detection via reinforcement learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 34, pp 516–523
34. Guo H, Cao J, Zhang Y, Guo J, Li J (2018) Rumor detection with hierarchical social attention network. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, pp 943–951
35. Ma J, Gao W, Mitra P, Kwon S, Jansen BJ, Wong K-F, Cha M (2016) Detecting rumors from microblogs with recurrent neural networks
36. Wang WY (2017) "liar, liar pants on fire": A new benchmark dataset for fake news detection. arXiv:1705.00648

37. Zhang S, He F (2020) Drcdn: learning deep residual convolutional dehazing networks. Vis Comput 36(9):1797–1808
38. CHEN Yuankun LJ (2020) Distantly supervised relation extraction with layered attention mechanism
39. Guo C, Cao J, Zhang X, Shu K, Yu M (2019) Exploiting emotions for fake news detection on social media. arXiv:1903.01728
40. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. arXiv:1706.03762
41. Vlad G-A, Tanase M-A, Onose C, Cercel D-C (2019) Sentence-level propaganda detection in news articles with transfer learning and bert-bilstm-capsule model. In: Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda, pp 148–154
42. Aggarwal A, Chauhan A, Kumar D, Mittal M, Verma S (2020) Classification of fake news by fine-tuning deep bidirectional transformers based language model. EAI Endorsed Transactions on Scalable Information Systems, 7(27)
43. Bawa VS, Kumar V (2019) Emotional sentiment analysis for a group of people based on transfer learning with a multi-modal system. Neural Comput & Applic 31(12):9061–9072
44. Kaliyar RK, Goswami A, Narang P (2021) Fakebert: Fake news detection in social media with a bert-based deep learning approach. Multimedia Tools and Applications 80(8):11765–11788
45. Rubin VL, Conroy N, Chen Y, Cornwell S (2016) Fake news or truth? using satirical cues to detect potentially misleading news. In: Proceedings of the second workshop on computational approaches to deception detection, pp 7–17
46. Jin Z, Cao J, Zhang Y, Luo J (2016) News verification by exploiting conflicting social viewpoints in microblogs. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 30
47. Shu K, Sliva A, Wang S, Tang J, Liu H (2017) Fake news detection on social media: A data mining perspective. ACM SIGKDD explorations newsletter 19(1):22–36

Changhao Tang is a postgraduate in University of Jinan, China from 2019. He won several first-class and second-class scholarships at school. He was awarded the Jinan Quancheng Scholarship. He is the President designate of CCF Student Chapter of University of Jinan. He has an invention patent under substantive examination. His research interests include natural language processing and data intensive computing.

Kun Ma received his Ph.D degree in Computer Software and Theory from Shandong University, Jinan, Shandong, China, in 2011. He is an Associate Professor in School of Information Science and Engineering, University of Jinan, China. He presides 1 National Natural Science Foundation of China (NSFC), and 5 provincial scientific research projects and 4 teaching research projects. He has authored and coauthored over 100 research publications in peer-reviewed reputed journals and conference proceedings. His entire publications have been cited over 680 times (Google Scholar). 7 of his papers have been cited over 20 times. The latest Google h-index of his publications is 14. He has served as the program committee member of various international conferences and reviewer for various international journals. He is the Co-Editor-in-Chief of International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM). He is the managing editor of Journal of Information Assurance and Security (JIAS). He is the editorial board member of Engineering Applications of Artificial Intelligence and International Journal of Grid and Utility Computing. His research interests include stream data processing, data intensive computing, natural language processing, and big data management. He has obtained 18 patents for inventions.

Benkuan Cui is a postgraduate in University of Jinan, China from 2020. His research interests include natural language processing and distributed computing.

Ke Ji received the Ph.D. degree in computer science and technology from Beijing Jiaotong University in 2016. He is an associate professor at University of Jinan. His research interests include machine learning, recommendation system.

Ajith Abraham received the Master of Science degree from Nanyang Technological University, Singapore, in 1998, and the Ph.D. degree in computer science from Monash University, Melbourne, VIC, Australia, in 2001. He is the Director of Machine Intelligence Research Labs (MIR Labs), Auburn, WA, USA, a Not-for-Profit Scientific Network for Innovation and Research Excellence connecting industry and academia. The Network with HQ, Seattle, WA, USA. He has currently more than 1000 scientific members from over 100 countries. As an Investigator/Co-Investigator, he has won research grants worth over U.S. $100+ Million from Australia, USA, EU, Italy, Czech Republic, France, Malaysia, and China. He works in a multidisciplinary environment involving machine intelligence, cyber-physical systems, Internet of Things, network security, sensor networks, web intelligence, web services, data mining, which are applied to various real-world problems. In these areas, he has authored/coauthored more than 1300+ research publications out of which there are 100+ books covering various aspects of computer science. One of his books was translated to Japanese and a few other articles were translated to Russian and Chinese. About 1100+ publications are indexed by Scopus and over 900+ are indexed by Thomson ISI Web of Science. Some of the articles are available in the ScienceDirect Top 25 hottest articles. He has 1100+ co-authors originating from 40+ countries. He has more than 41 000+ academic citations (Hindex of 95 per Google Scholar). He has given more than 100 plenary lectures and conference tutorials (in 20+ countries).