Applied Intelligence (2023) 53:8354–8369
https://doi.org/10.1007/s10489-022-03910-9

DC-CNN: Dual-channel Convolutional Neural Networks with attention-pooling for fake news detection

Kun Ma[1] · Changhao Tang[1] · Weijuan Zhang[2] · Benkuan Cui[1] · Ke Ji[1] · Zhenxiang Chen[1] · Ajith Abraham[3]

Accepted: 18 June 2022 / Published online: 1 August 2022

Abstract

Fake news detection mainly relies on the extraction of article content features with neural networks. However, it has brought some challenges to reduce the noisy data and redundant features, and learn the long-distance dependencies. To solve the above problems, Dual-channel Convolutional Neural Networks with Attention-pooling for Fake News Detection (abbreviated as DC-CNN) is proposed. This model benefits from Skip-Gram and Fasttext. It can effectively reduce noisy data and improve the learning ability of the model for non-derived words. A parallel dual-channel pooling layer was proposed to replace the traditional CNN pooling layer in DC-CNN. The Max-pooling layer, as one of the channels, maintains the advantages in learning local information between adjacent words. The Attention-pooling layer with multi-head attention mechanism serves as another pooling channel to enhance the learning of context semantics and global dependencies. This model benefits from the learning advantages of the two channels and solves the problem that pooling layer is easy to lose local-global feature correlation. This model is tested on two different COVID-19 fake news datasets, and the experimental results show that our model has the optimal performance in dealing with noisy data and balancing the correlation between local features and global features.

Keywords COVID-19 · Fake news · Noisy data · Convolutional neural network

1 Introduction

Nowadays, mobile social media is an Internet-based form of communication for people to express their opinions online [9]. Social media is a double-edged sword as an integral part of society and culture. On the one hand, the openness and convenience of its platforms offer people the space to express themselves freely [30]. On the other hand, it also facilitates the spread of fake news [32]. It is shown that social media is a significant channel to spread numerous unfiltered content, as well as a place for misinformation and fake news transmission [1]. The misuse of social media news affects the mental health of users leading to spread of public panics [9].

COVID-19 has been swept across the world since being announced in 2019. As of February 1, 2022, more than 370 million people got infected. With the outbreak of a pandemic, fake news and misinformation have become rampant on social media either. Fake news about COVID-19 even spreads more rapidly than real news [21]. Under such circumstances, it has not only created public panic about the epidemic, but threatened public mental health [1]. On social media platforms, the circulation of unverified misinformation will put people at great risk. For example, a piece of fake news about "Drinking bleach can cure coronavirus disease" led to mass mortality [3]. However, there is no effective way to verify the authenticity of content shared on social media platforms. Thus, it is urgent to figure out how to detect and govern fake news [21].

Fake news on social media carries unique characteristics, which pose great challenges for fake news detection. First, fake news often deliberately misleads readers, causing difficulties for news contents detection [30]. Second, social media news contains noisy data and redundant features for lacking standard and rigorous format [14]. Third, content features are highly context-dependent and ambiguous, which increases the difficulty of accurately extracting key features from the model [34].

It is shown that sequence neural networks are widespread for fake news detection to encode news content and

✉ Kun Ma
ise_mak@ujn.edu.cn

Extended author information available on the last page of the article.

social contextual information [13]. As one of the most common detection models, Convolutional Neural Network (CNN) architecture is proposed considering both local and global features. However, CNN is unable to capture long-distance dependencies [33], and its pooling layer cannot keep the correlation between local features and global features [28]. Recently, word embedding technology has combined with CNN. Taking bow-CNN for example, it replaces the convolution layer with the BoW model, integrating word order information into feature vectors [10]. The improved Word Vectors (IWV) utilize pre-trained words embedded in combination with different modules of CNN [25]. It enhances the word embedding model through lexical, positional, and syntactic features. Nevertheless, those models have the challenges of learning long-distance dependencies and and the importance of different words [4]. Meanwhile, existing word embedding technologies, such as Fasttext and Word2vec, fail in taking an impelling solution to reduce redundant data. Attention mechanism of Transformer [31] is proposed to learn long-term dependencies effectively. But such a mechanism is inferior to RNN and CNN in obtaining local information when comprehending short text features. Current methods are solely effective to learn one of the local features and global features, and cannot simultaneously extract two features.

To address all the above issues, this paper has proposed a word embedding method for Chinese text and a detection model of Dual-channel CNN with Attention Pooling(DC-CNN). A dynamic word embedding method is proposed to reduce noise data and semantic ambiguity. The Attention mechanism is further added to the pooling layer to tackle the simple loss of feature correlation in traditional CNN. Due to its excellent performance in learning long-distance dependencies, it can also strengthen the learning of global features and key features.

Our main contributions are summarized as follows:

- **Dynamic word embedding for Chinese text (abbreviated as DWtext).** DWtext is the dynamic word embedding method proposed by DC-CNN. It performs word segmentation on social media data according to Chinese syntactic characteristics. By filtering the redundant parts of word segmentation, it cannot only retain the features useful for classification, but also reduce the noise data. The adjacent word segmentation is expressed in the form of "N-gram" and related to each other, allowing the model to learn the order information between words. The conditional probability of intermediate word vectors is utilized to predict contextual word vectors, so that word vectors with the same context have similar semantics. Compared with traditional static word embedding, DWtext can dynamically generate word vectors with different contexts, which can reduce semantic ambiguity in a more effective way. Therefore, corresponding word vector can be generated by the model even confronted with words exclusive the training vocabulary.
- **Dual-channel CNN with Attention Pooling (abbreviated as DC-CNN).** DC-CNN replaces the traditional pooling layer with a Dual-channel pooling layer, which incorporates with a Max-pooling layer and an Attention-pooling layer. The Max-pooling layer can remarkably decrease redundant features by allowing neurons in one layer to solely focus on active features while ignoring inactive ones. However, it is also followed by the easy loss of dependencies between local features. In that case, attention-pooling is adopted as a remedy. It has a great advantage in capturing long-distance dependencies, which provides a natural advantage of our model to learn global semantics. Thus, DC-CNN, combining the benefits of the two channels, can effectively compensate for the lack of correlation between local features and global features.

The remainder of the paper is organized as follows. The related work on fake news detection is discussed in Section 2. Section 3 presents the problem definition of fake news detection. Section 4 presents the architecture, DWtext, Dual-Channel pooling layer of our DC-CNN framework. The datasets of are described in Section 5. Section 6 shows the experimental results on four datasets with detailed analysis. Finally, brief conclusions and future research directions are outlined in the last section.

2 Related works

2.1 Word embedding of fake news detection

Many fake news detection algorithms hinge on extracting statistical and semantic features from news content [8]. But as a humanly abstract expression, language cannot be directly recognized by computers. Therefore, the first step in Natural Language Processing (NLP) is to encode the unstructured character data. And such a step aims at obtaining the mapping relationship between text and digital space [24]. The limited data representation of texts is a bottleneck constraining fake news detection [29]. In the traditional way, text vectorization can be easily achieved by one-hot encoding. But its encoded vector can neither reflect the similarity and connection between words nor contain any semantic information. Thus, the method based on the bag-words-model summarizes and analyzes the frequency of individual words or "n-grams" to extract deceptive information [29]. But this representation greatly hinges on independent N-gram, and meanwhile is separated from the context. As a distributed representation, word embedding maps each word to a low-dimensional and dense eigenvector, with real numbers in each dimension [27]. Sentences with various lengths can be represented by tensors of different dimensions, solving the problems of the sparse and high-order features beyond traditional methods. After combining with the neural network, word embedding has been generally employed in Natural Language Processing (NLP) tasks. While acquiring contexts, Word2vec can generate word vectors with contiguous space [14]. It proposes two various model architectures, Skip-Gram and CBOW [20]. Among them, the former draws on middle words to predict neighbor, and the other utility context words to predict middle words. Subsequently, a series of word vector technologies such as GloVe [22] and FastTest [12] are proposed to provide a variety of numerical representation methods for the text. However, these methods have challenges to deal with redundant data to improve the accuracy of feature learning.

Although long-distance dependencies cannot be learned, CNN is often used to extract local features, especially when local features in long texts are very important [4]. Bow-CNN [10] replaces the convolution layer with BoW (Bag-of-words) model and uses word order to improve the performance of topic classification. Improved Word Vectors (IWV) [25] utilizes pre-trained word embedding in combination with CNN, but has the challenges to capture long correlation and the importance of different words.

2.2 Fake news detection

When detecting fake news, it is difficult to distinguish fake news posts from real ones through various patterns of information like text content and social context [23]. For fake news detection, there are traditional machine learning [26] and deep learning based models [26]. Traditional methods are primarily reliant on the artificial extraction of news content features from statistical features to semantic features [35]. And those features are fed to machine learning classifiers for learning, such as naive Bayes classifier, Support vector machine (SVM), decision tree, etc [14]. However, as language patterns are highly reliant on specific events and domain knowledge, it is arduous to extract text features manually [36]. Meanwhile, these features are limited to learn complicated patterns, resulting in poor generalization performance in tasks of fake news detection [23]. Consequently, deep learning models are proposed to learn text features in time series with constructing neural networks [35]. Unlike traditional machine learning, these deep ones are deft at learning content features automatically, making a huge stride in fake news detection [36]. In 2016, Recurrent neural network (RNN) is proposed to model the content of the article as a continuous-time series than the traditional way [19]. Convolutional neural network (CNN) is applied to text classification tasks, using layers with filters to extract local features from text [24]. However, limited by the size of the convolution kernel, CNN failed to capture the long-term dependence relationship. Bidirectional GRU can make up for the deficiency of CNN in extracting semantic information from long text, but it is inferior to CNN in extracting local features [5]. In that case, DPCNN proposed a low-complexity word-level deep CNN for text classification, enhancing the comprehension of long-term dependencies [11]. But the fixed number of feature maps in DPCNN restrains the deep learning of semantic space. Though the deficiency for CNN in learning long-term dependence is balanced to some extent by RCNN through enhancing the model's learning of sequence data [15], the gradient disappearance in RNN still lies on the table.

In essence, the attention mechanism serves as automatic learning of attention weight matrix [2]. It assigns different weights to different words in the source sentence, and predicts the target words through these words, so as to strengthen the relationship between the target sentence and the source sentence. Simultaneously, the attention matrix is visualized to know which part of data is concerned in the training process of the neural network. In addition, learning of long-distance dependency and key features are heightened as the attention mechanism introduced. A social attention network was also proposed to capture the hierarchical structure of microblog events [7]. Attentional mechanism is to integrate the social situation into a hierarchical bi-directional long short-term memory model. Although the Attention mechanism extols in acquiring global information by its advantage of learning long-term dependencies, it is inferior to RNN and CNN in local information obtaining facing with short-text.

Recent deep models also have some drawbacks [4]. Attention-based Multichannel Convolutional Neural Network (AMCNN) [18] combines the learning advantages of LSTM and CNN, while applying attention mechanism to the hidden states of Bi-LSTM. Although making better use of word order information, it still fails to work out the loss of temporal order information. Combining with Bi-LSTM, attention mechanism and convolution layer, Attention-based Bidirectional Long Short-term Memory with Convolution Layer (AC-BiLSTM) [17] is capable of capturing both local features of phrases and global sentence semantics. But it does not solve the problem of co-occurrence of both short and long dependencies [4]. Therefore, our method emphasizes how to make the classifier either keep its advantage in learning local features or enhance its learning in long-term dependencies.

There are some recent detection methods on fake images [16], such as MTD-Net [38] and MTD-Net [39]. Attention mechanism is also applied in the rumor detection process. These methods are designed for image detection, but the text detection of fake news is not discussed. Compared with MTD-Net and MTD-Net, our DC-CNN focuses on the attention mechanism of text classification.

3 Problem definition

3.1 Fake news and fake news detection

Fake news is defined in two various aspects [41]: Broadly speaking, "fake news" is defined as "false news" with more emphasis on the authenticity of information while less on the influence of its intention. In the narrow sense, fake news refers to those intentionally fabricated but provably false articles [42]. Narrow definition is adopted in this paper with the following reasons: First, the intention behind fake news are proved to be of great significance in guide the research and identification of the epidemic fake news [30]. Second, a detection technique under a narrow definition can also be applied in a broad sense.

Fake news detection aims to detect the credibility of given news. In this paper, fake news detection is treated as a binary classification problem since the new crown fake news in social media essentially represents a subreption by the publisher. The classification label set is represented as $Y = \{0, 1\}$, where 0 represents true news and 1 represents fake news. We want to establish a mapping between news and labels as $\mathcal{F} : Y \rightarrow \{0, 1\}$. Given news A, the target of the fake news detection model is to obtain its corresponding classification label:

$$\mathcal{F}(A) = \begin{cases} 0, & \text{if } A \text{ is a piece of true news} \\ 1, & \text{if } A \text{ is a piece of fake news} \end{cases} \quad (1)$$

3.2 Architecture of fake news detection

There are two components of our model in logical, as shown in Fig. 1:

(1) **Text vectorization:** After received by the model, text data will be encoded into a recognizable vector representation by computer. The first layer of the model is called the embedding layer in which we transform the given text into accurate feature vector by DWtext method. Noisy data and redundant features have very much on the decline, leading to a more effective learning in classification features.

(2) **Classification:** This part equips the model withthe ability of automatic classification by capturing the feature information in the vector representation. The second layer of the model is the convolution layer, where three different convolution kernels are used to capture local dependencies between adjacent words. The feature vectors are transformed into the corresponding feature matrices through this step. The dimension of the word embedding vector is set to 300 in our model, the filter sizes are 3,4 and 5 respectively, with each of which contains 128 feature maps.

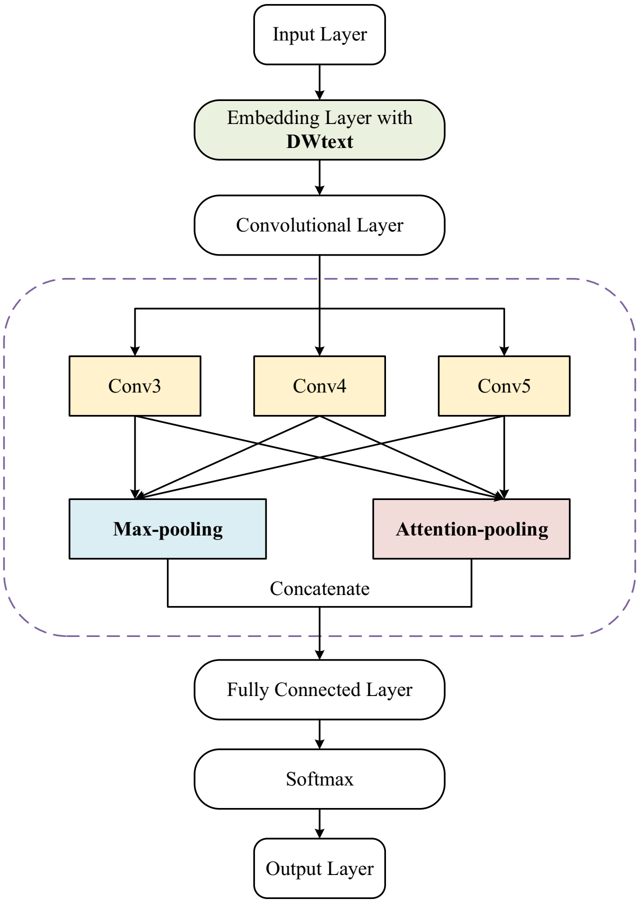


Fig. 1 General architecture of fake news detection

The third layer is the pooling layer. The three feature matrices from convolution layer are input into the two different pooling layers respectively to learn the classification features. One of the channels names Max-pooling layer, whose max pooling operation can extract the most valuable information of local features but discard those useless and irrelevant ones, resulting in feature vectors with fixed dimensions. Another channel, Attention-pooling layer introduces the Attention mechanism, aiming at compensating for the dependencies in long-distance learning of CNN. Then the output vectors of the two channels are connected and fed into subsequent neural network layer for classification. In this process, we combine both the advantages of Max-pooling layer in learning local relations and of Attention-pooling layer in learning global relations.

4 Methods

4.1 Architecture

The structure of this network can be divided into four parts: Embedding Layer, Convolutional Layer, Dual-pooling Layer, and Classifier. Firstly, Embedding Layer with DWtext can generate word embedding after text preprocessing such as text cleaning and text tokenization. Next, Convolutional Layer extracts the text features Preliminary. In Dual-pooling layer, Max-pooling layer capture local feature, Attention pooling layer capture long distance dependency. Finally, the final text embedding is input into the classifier to generate and visualize the classification results after connecting two vectors generated by Max-pooling layer and Attention pooling layer respectively. The detail is shown in Fig. 2

4.2 DWtext

DWtext is the dynamic word embedding method proposed by DC-CNN. It can not only reduce noisy data and redundant features, but also reduce semantic ambiguity. DWtext can help our model handle new derived words better than traditional models at the same time.

The characteristics of the data were analyzed as follows. It is obvious that the Chinese word segmentation system is different from that of English. Chinese words is divided according to their semantics, and words are linked together without a unified delimiter such as space. Meanwhile, Chinese social media news presents unique characteristics [30]. With no strict censorship available in social media platform, self-media news is invariably rife with problems like frivolous content and irregular form. The news from individual users in particular, incorporates with numerous emoji, symbols and other noise data, which will provoke error of the model in learning classification features if directly applied in model training without process. Therefore, we perform word segmentation and data cleaning on Chinese data to retain the information valid for classification in the data.

English words contain many prefixes and suffixes. The etyma and affix in English contexts usually have specific meanings and can be combined with other words to form new words, such as “ing” for primary tenses. N-gram can be used to effectively learn the combination features between etymas and words.Different from English words, Chinese characters is composed of radicals. In that case, we divide the data into different N-grams based on Chinese characters to learn the complete Chinese syntactic features.

Based on the above analysis,, the dynamic word embedding method for Chinese text was proposed in this paper to create a data dictionary, as shown in Fig. 3. The specific process is as follows:

Firstly, word graph scan is performed on the text data based on prefix dictionary to obtain all segmentation results. Secondly, a directed acyclic graph (DAG) is constructed according to the different segmentation positions. Again, the final word segmentation is obtained by calculating the maximum probability path. In addition, the Hidden Markov

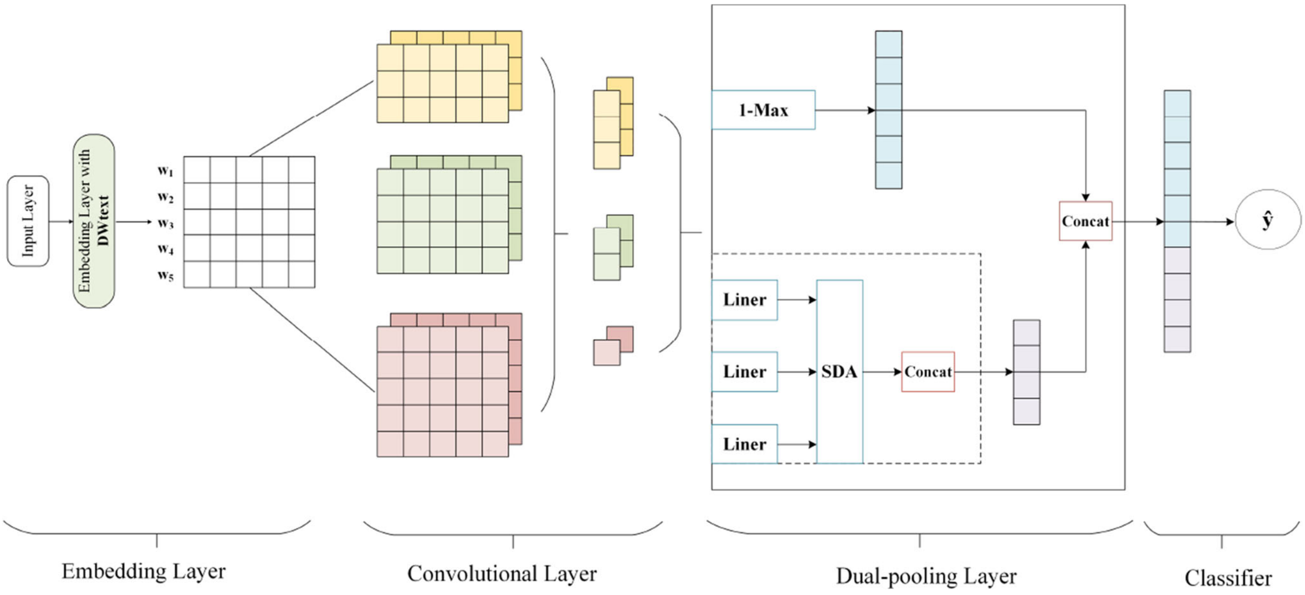
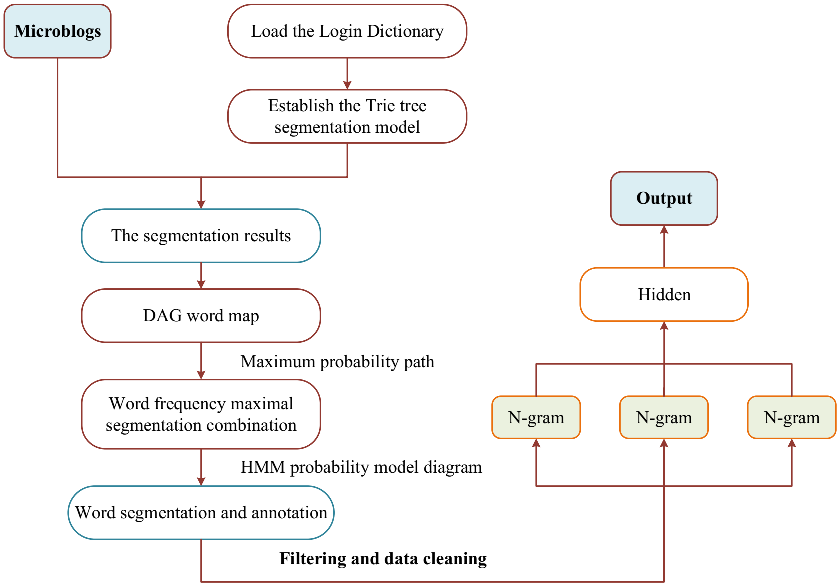


Fig. 2 Architecture of DC-CNN

Fig. 3 Dynamic word embedding for Chinese text


Microblogs
Load the Login Dictionary
Establish the Trie tree segmentation model
The segmentation results
DAG word map
Maximum probability path
Word frequency maximal segmentation combination
HMM probability model diagram
Word segmentation and annotation
Filtering and data cleaning
N-gram
N-gram
N-gram
Hidden
Output

Model(HMM) is used to learn the words that are exclusive in the word segmentation lexicon. After the elimination of the noise data from special characters, emoticons to garbled characters in the segmentation, the cleaning word segmentation is obtained. The word segmentation after filtering were recombined into a whole and then divided into N-gram sub-words for training. As the word order information is incorporated into the generated word embedding, the sparsity problem in dictionary embedding also comes to be settled [40]. The conditional probability of intermediate word vector W_t is used to solve the context word vector, and the formula is as follows:

$$P\left(W_i \mid W_t\right)=\frac{P\left(W_t * W_i\right)}{P\left(W_t\right)} \tag{2}$$

Where $i = t-1, t-2, t+1, t+2$, W_t is the target word vector, W_i represents the context vector.

Finally, the matrix of input sentence is obtained as represented by $\mathrm{S} = \mathrm{R}^{\mathrm{n*d}}$, where n is the number of words in the sentence and d represents the dimension of word embedding. The influence caused by redundant features on classification is eliminated, with the generated dynamic word vector being more effective in ambiguity decreasing.

4.2.1 Convolution layer

Traditionally, CNN is mainly composed of convolution layer, pooling layer and full connection layer. In our model, both the traditional convolution layer and full connection layer were retained. For the input sentence matrix $\mathrm{S} = \mathrm{R}^{\mathrm{n*d}}$, DC-CNN proceeds convolution in the way of linear filters. Let the weight matrix of the convolution kernel be $\mathrm{W_c} = \mathrm{R}^{\mathrm{h*d}}$, where c is the number of convolution kernels, h is the width of the convolution kernels, and d is the length of the convolution kernels. It is worth noting that the value of d is equal to the dimension of the word embedding vector in our model. Through a series of convolution operations we get the feature-mapping vector $\mathrm{O} = \left[\mathrm{o}_0, \mathrm{o}_1, \ldots, \mathrm{o}_{\mathrm{n-h}}\right] \in \mathrm{R}^{\mathrm{n-h+1}}$ of sentence matrix S. o_i is the submatrix of convolution kernel matrix W and S, and the formula is:

$$o_i = W \cdot S_{i:i+h-1} + b \tag{3}$$

Where $i = 0, 1, 2, \ldots, n-h$, $(\cdot)$ represents the dot product of the matrix, $S_{i:j}$ represents the sub-matrix of the S-matrix from i rows to j rows, as well as the word embedding matrix from the i-th word to the j-th word in the sentence, and b is the bias term.

After convolution, the mapping vector O of each feature is input into the dual-channel pooling layer for feature extraction and filtering.

4.3 Dual-Channel pooling layer

In order to address the deficiency of relevance between local features and global features in CNN, this article adopts Dual-channel Pooling Layer rather than the traditional one. The pooling layer we adopted consists of a Max-pooling Layer and an Attention-pooling Layer, as shown in Fig. 4. The learning advantages of the two methods are combined perfectly as the feature matrices from different channels stitched as a whole.

4.3.1 Max-pooling layer

After filtering those convolutional feature vectors, the Max-pooling layer can extract the active part of its local features as well as channel long sentences into fixed length. As model overfitting can be largely decreased along with the reducing redundant features, the Max-pooling layer makes a great stride in the robustness of the model by the sustaining the most valuable convolutional results in fake news detection. The Max-pooling layer in this paper is regarded as one channel of the dual-channel pooling layer. The process in details is shown in Fig. 4. Filters with various region sizes are further applied in this essay to obtain multiple 1-max pooling values. After the Max-pooling layer, we can get the most important feature v in each feature-mapping vector O, and the formula is as follows:

$$v = \max_{0 \le i \le s-h} \{o_i\} \tag{4}$$

The 1-max pooling values from the feature mapping are connected to the output of the last hidden layer of the attention-pooling layer. Through the above operations, the most important local features are extracted into the final result.

4.3.2 Attention-pooling layer

The Attention-pooling layer is proposed as another channel of the pooling layer for the two following reasons: Firstly, by virtue of its advantage as learning long-term dependencies of sentences, the Attention mechaism can availably remedy CNN's deficiency in capturing global features. Secondly, as the semantical importance of words varies from sentences to sentences, the Attention mechanism proved to be effect in extracting the semantic information of important words.

The multi-attentional mechanism was mainly adopted in this channel. Compared with the traditional one, the multi-attentional mechanism is more comprehensive in attention information gaining. It can obtain comprehensive attention information from multiple dimensions by performing different linear transformations of the input feature matrix [6]. Similar to CNN in learning with different filters, multi-attentional mechanism is also a combination of multiple self-attentional mechanisms, with each of them incorporates three main transformation matrices: Query(Q), Key(K) and Value(V). In the initial stage, the matrix Q, K and V is equivalent to the global feature matrix O of the convolutional layer output, as shown in the following formula:

$$Q = K = V = O \tag{5}$$

Scaled Dot-product Attention (SDA) is the main algorithm of the self-attention mechanism. Firstly, take the inner product of Q and each K separately to calculate their similarity. Secondly, softmax is used to calculate the weight after dividing the similarity result by the square root of d, where d_k is the dimension of K. Finally, the weight obtained is multiplied by value to obtain the output $SDA(Q, K, V)$ from the attention layer, as shown below.

$$\mathrm{SDA}(Q, K, V) = \mathrm{SoftMax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{6}$$

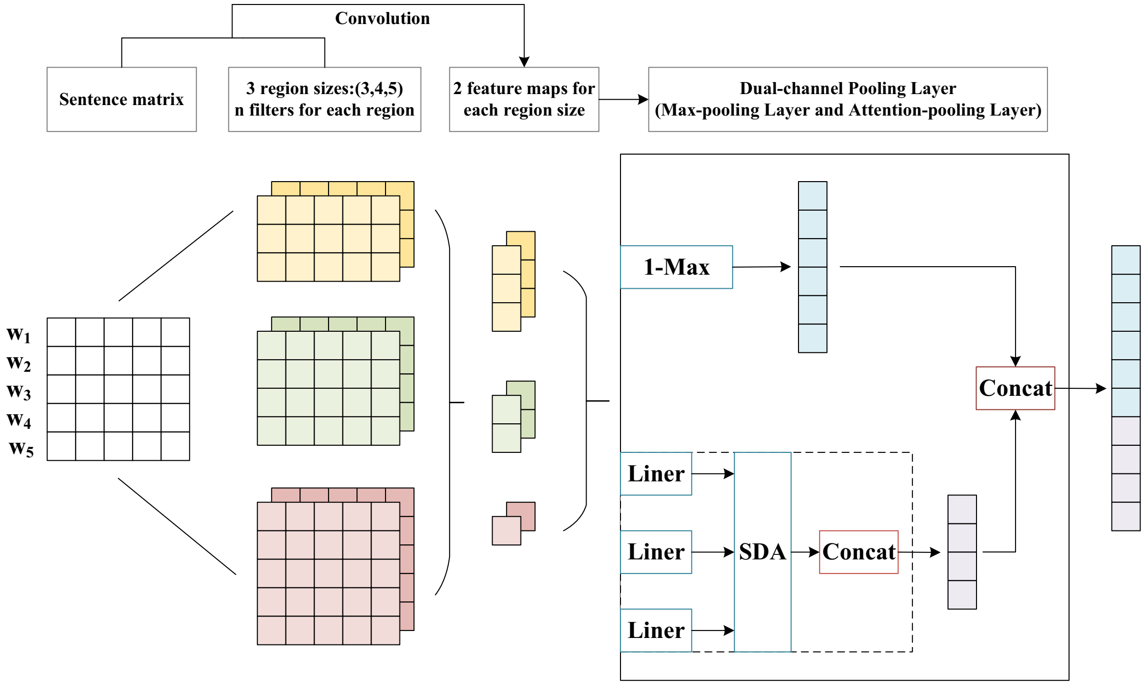


Fig. 4 Dual-channel Pooling Layer

Multi-attentional mechanism performs many different linear transformations on the feature vector matrix. In each linear transformation, different weight matrices W_i^Q, W_i^K and W_i^V are used to obtain a variety of representations. Then, the SDA output results from h different self-attention modules are concatenated into a whole. In this paper we set the value of h to 8. At last, the final result is converted to an output vector of fixed dimensions through the matrix W_o transformation, and the formula is as follows:

$$\text{head}_i = \text{SDA}\left(QW_i^Q, KW_i^K, VW_i^V\right) \tag{7}$$

$$\text{Head}\left(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}\right) = Concat\left(head_1, \cdots, head_h\right) W^{\circ} \tag{8}$$

The two feature matrices from the Max-pooling layer and the Attention-pooling layer are spliced into a whole. Therefore, the model can learn important local features and global feature information in the text at the same time.

5 Dataset

We conducted experiments with two different datasets to test the generalization ability of DC-CNN, the details are shown in Table 1. Both data sets are fact-checked social media data, so the experimental results obtained are highly reliable. The two datasets are described in detail below, which can be available at https://github.com/SmallZzz/FakeNewsData.

COVID-19 Dataset based CHECKED(abbreviated as COVID-19): CHECKED is the first Chinese dataset on COVID-19 misinformation [37]. It contains 2,120 pieces of microblog posted from December 2019 to August 2020, of which 344 were "fake news" and 1,776 were "true news". Each data has been strictly reviewed and officially certified, and contains the following fields: id, label, date, user_id, user_name, text, pic_url, video_url, comment_num, repost_num, like_num, comments and reposts. However, the quantity of fake news in this data is small, which easily leads to the over-fitting problem of the model. So we added another 718 pieces of data from Tencent's fact-checking platform. We reshuffled the combined Data to form COVID-19 Data. It contains 2,838 pieces of data, of which 883 are "fake news" and 1,955 are "true news".

BAAI Dataset: Beijing Academy of Artificial Intelligence(BAAI) and the Institute of Computing Technology of the Chinese Academy of Sciences jointly held a fake news detection competition on the Internet in 2019. The dataset for this competition was provided by BAAI and is of high quality. It contains 38,471 pieces of data, including 19,285 being "Fake" and 19,186 being "True". Each piece of data is represented by the following components: id, text, and label. the length of news in this dataset mostly concentrates within the range of 120-150.

6 Experiments

The generalization ability of a model refers to the ability of a pre-trained model to handle unseen data, which is mainly related to the complexity and training of the model. Excellent prediction models can be generalized across data sets and tasks and show strong applicability. Many current studies use a single dataset for training and typically evaluate their performance on different subsets of the same dataset. This approach leads to inadequate conclusions. Therefore, this paper uses two different data sets for experiments.

In our experiment, fake news detection is regarded as a binary classification task, in which fake news is regarded as a positive category and true news as a negative category. Precision(P), Recall(R), F1-score($F1$) and Accuracy(ACC) were used to evaluate the performance of the model. We divided the dataset into training set, validation set and testing set in a ratio of 3:1:1. The following two sections describe the analysis of the results when the model uses the COVID-19 dataset and the BAAI dataset respectively.

To specify the experimental process, we detailed the configuration of baseline models. The similarities of the configuration of our model and baseline models are: all models use DWtext method to create thesaurus; the epoch of training all samples of all models is 20; the word embedding dimension EMBEDDING_DIM of all models is 100; the batch parameter batch_size is 128. The differences of the configuration of our model and baseline models are shown in Table 2.

The configuration of hardware and software is shown in Table 3. The running OS is Windows 10 x64, the version of Python is 3.6.5, the version of tensorflow is 1.12.0, the

Table 1 Dataset introduction

Dataset	Category	Average length	Fake/True number	Training/Validation/Testing number
COVID-19	2	344	883/1955	1702/568/568
BAAI	2	300	19285/19186	23082/7694/7695

Table 2 Configuration of DC-CNN and baseline models

Model	Detailed configuration
TextCNN	Convolution kernel = [3,4,5]
	Feature maps = 100
	Pooling layer = Max-pooling layer
Multi-channel CNN	Pooling layer 1 = Max-pooling layer (updated)
	Pooling layer 2 = Max-pooling layer (not updated)
RCNN	Pooling layer = Max-pooling layer
	hidden_size of LSTM = 256
DPCNN	Pooling layer = Max-pooling layer
	num_filters = 200
Transformer	Attention head = 8

version of Keras is 2.2.4. The CPU of the device is I7-8750H, the GPU is GeForce GTX 1060, and the memory is 16 GB.

The main contents of this chapter are as follows: Firstly, this chapter compares the experimental results of different models and analyzes the results. Secondly, the DWtext method is compared with different word embedding methods. Finally, the effect of different channels in DC-CNN was tested through ablation experiment, and the experimental effect of the model on different length data sets was tested.

6.1 Overall model effect

6.1.1 Experimental results of COVID-19 datasets

In the text branch, we experimented with a variety of models using COVID-19 datasets, and the results are as shown in Table 4. The experimental results show that textCNN has a good effect on fake news detection, and its F1 value is 89.26%. Multi-channl CNN performed better, with a F1 value of 92.18%. Multi-channl CNN replaces the traditional pooling layer with the multi-channel pooling layer. Compared with textCNN, multi-channel CNN can learn more local features, so its precision and recall rate are improved to a certain extent. However, it still fails to solve the problem of global feature loss caused by pooling layer. At the same time, long-term dependence cannot be learned under the limitation of convolution kernel size. Transformer has a high recall rate, but has a low precision rate of 82.99%, which is lower than most models. Although it can effectively learn the long-term dependence relationship, it still lacks the ability to learn the local key features. DC-CNN enhances global feature learning by introducing attention mechanism. Meanwhile, the advantages of the classifier for local feature learning are maintained by using Max-pooling layer. Its F1 value is 94.81%, which is 20.24% higher than RCNN, which has the worst effect. It is worth noting that even if the number of true and fake news in the dataset is unbalanced, DC-CNN still maintained the highest precision and recall rate at the same time. This shows that it can most accurately learn the features that are critical and effective for classification. The effect of RCNN and DPCNN is not ideal, RCNN is even lower than 80%. It is not difficult to see that the recall rate of these two models is very low. This shows that although they can identify some fake news, they also misjudge many fake news as true news. In the case of small datasets, both the two models overfit simultaneously. Compared with DC-CNN, they cannot effectively learn the key features related to classification. It can be seen from the experimental results that BERT and ERINE had not achieved better results than DC-CNN. For specific type of dataset such as

Table 3 Environment configuration

Hardware and software	Configure
CPU	I7-8750H
GPU	GeForce GTX 1060
RAM	16 GB
Operating System	Windows 10 x64
Environment	Python: 3.6.5
	Tensorflow: 1.12.0
	Keras: 2.2.4

Table 4 Experimental results of COVID-19 datasets

Model	P	R	ACC	F1
textCNN	0.8926	0.8926	0.8926	0.8926
Multi-channel CNN	0.9194	0.9243	0.9217	0.9218
RCNN	0.8116	0.7204	0.7457	0.7457
DPCNN	0.8690	0.7683	0.8995	0.8155
Transformer	0.8299	0.8908	0.8592	0.8586
BERT	0.9281	0.8820	0.9470	0.9045
ERNIE	0.8771	0.9752	0.9541	0.9235
DC-CNN	0.9473	0.9489	0.9481	**0.9481**

The bold parts are the optimal results of different experiments

COVID-19 datasets that are composed of COVID news, pre-trained models perform weakly in fake news detection for lacking large-scale annotated data and sufficient semantic understanding of domain-specific knowledge.

6.1.2 Experimental results of BAAI datasets

To avoid the model showing good performance only for specific datasets, we conducted a second experiment using BAAI datasets while maintaining the same model configuration, and the results are as shown in Table 5. The experimental results show that DC-CNN has the highest values of F1-score and Accuracy, and has stronger universality and portability. Transformer has the lowest F1-score at 83.50%. It also had the lowest recall rate at 81.32%. Combining the results of the two experiments, it can be seen that Transformer is inferior to CNN in local feature extraction. It is easy to overfit in the learning process, which makes the model learn wrong classification features. Therefore, in the face of datasets with fewer data items and unbalanced classification, it is easy to amplify the disadvantage of short-text data learning. BERT and ERNIE can carry out bidirectional training, which is better than Transformer in feature extraction, but still worse than DC-CNN. The F1-score values of CNN and DPCNN are close, 90.59% and 90.68% respectively. By observing the precision and recall rates of the two models, it can be seen that the fake news is more likely to be predicted by CNN in the dataset, while the right proportion of the fake news predicted by DPCNN is higher. The number of feature map is fixed in DPCNN, so that the operation of adjacency merging is carried out in the original space or the space similar to the original space. Although the entire network architecture is deep, it is actually flat from a semantic space perspective. Therefore, Both DPCNN and CNN are relatively weak in learning long-term dependency relationship. DC-CNN also enhances the learning ability of global features and local features, enabling it to maintain excellent feature extraction ability in the face of social media data with a large amount of data noise. Experiments show that DC-CNN can achieve a high recall rate as well as a high precision rate, indicating that it has a stronger learning performance for the accuracy and integrity of classification features.

Based on the above experiments, the following conclusions can be drawn: Firstly, DWtext can better process Chinese social media data and extract rich context information. Secondly, DC-CNN can effectively deal with the correlation between local features and global features, and accurately extract key features related to classification. Finally, DC-CNN has strong robustness and universality and can adapt to different environments.

Table 5 Experimental results of BAAI datasets

Model	P	R	ACC	F1
textCNN	0.9059	0.9059	0.9059	0.9059
Multi-channel CNN	0.9774	0.9780	0.9777	0.9777
RCNN	0.8886	0.8651	0.8761	0.8767
DPCNN	0.9218	0.8923	0.9066	0.9068
Transformer	0.8621	0.8123	0.8882	0.8350
BERT	0.9541	0.9771	0.9647	0.9655
ERNIE	0.9705	0.9645	0.9673	0.9675
DC-CNN	0.9812	0.9819	0.9815	**0.9816**

The bold parts are the optimal results of different experiments

6.2 Comparison of word embedding models

Word embedding is an important factor affecting model performance. The distributed representation method of word has been widely used in text classification, which can integrate semantic information into word vector. The common methods include fastText, CBOW and Skip-Gram. We use the control variable method to test the influence of different word embedding methods on DC-CNN classification. In the experiment, we only changed the embedding layer of DC-CNN, and all other model parameters remained unchanged. We tested the effects of different methods on COVID-19 datasets and BAAI datasets, as shown in Tables 6 and 7. Experimental results show that DWtext can obtain optimal results on both datasets.

Fasttext model performed poorly on both datasets, with experimental results 2.57% lower than DWtext on Covid datasets and 0.39% lower than DWtext on BAAI datasets. This is because the Fasttext model is built from discrete unit words without any genetic attributes to each other, and therefore has a natural defect for learning continuous semantics. Besides, to avoid creating too much parameter space, the sliding window N is limited in size, so Fasttext can learn a limited amount of context information only. The Skip-Gram model and CBOW model perform better than Fasttext. They are two different model architectures for Word2Vec that can generate word vectors with continuous spatial information. Although the learning performance

Table 6 Experimental results of different word embedding methods (COVID-19 datasets)

Method	P	R	ACC	F1
Fasttext	0.9241	0.9489	0.9225	0.9224
Skip-Gram	0.9299	0.9331	0.9313	0.9315
CBOW	0.9332	0.9331	0.9331	0.9331
DWtext	0.9473	0.9489	0.9481	**0.9481**

The bold parts are the optimal results of different experiments

Table 7 Experimental results of different word embedding methods (BAAI datasets)

Method	P	R	ACC	F1
Fasttext	0.9775	0.9778	0.9776	0.9777
Skip-Gram	0.9805	0.9801	0.9803	0.9803
CBOW	0.9808	0.9806	0.9807	0.9807
DWtext	0.9812	0.9819	0.9815	**0.9816**

The bold parts are the optimal results of different experiments

is better than the FastText model, they also have some problems. We can also find that the average length of the two data sets is relatively large. Although the long text can extract more features than the short text, it also contains more redundant words. Most of these words are irrelevant words that have no positive influence on the classification results, which can easily reduce the accuracy of classification. The two models have no effective way to deal with the noisy data, and the learning ability of new derived words is insufficient. Compared with the model based on BoW, DWtext method can obtain more context information and solve the problem of data sparsity. At the same time, it can effectively eliminate the data noise in Chinese media.

6.3 Additional experiments

6.3.1 Ablation experiment

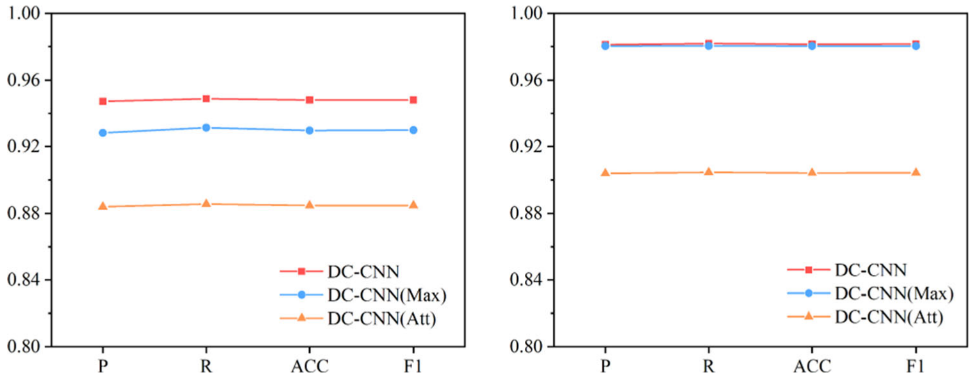
We examine the effectiveness of Max-pooling layer and Attention-pooling layer. Two variants for DC-CNN are developed, DC-CNN(Max) and DC-CNN(Att), which is DC-CNN only using max-pooling layer and attention-pooling layer. The comparison results presented in Fig. 5a and b. Meantime, the distribution of the text length of different data sets is analyzed to better explain the experimental effect of the model, as shown in Fig. 6a and b. It should be noted that the text length here is the data token length after text preprocessing (such as text cleaning and text tokenization).

Ablation experimental results of COVID-19 datasets: Fig. 5a illustrate that DC-CNN has the best performance, and DC-CNN(Max) always show better performance than DC-CNN(Att) on COVID-19 datasets. For the poor performance in DC-CNN(Max) , it is due to that this model only learns the local information between adjacent words. The lacking on global information interaction leads to the difficulty to transfer the text information of each word to others. It results in discontinuous information of the entire text. For the worst performance in DC-CNN(Att), it is due to that this model ignores the local features although the attention-pooling layer can capture the long-term dependencies of sentences.

Max-pooling layer has the advantage of capturing the local features, such as key words and and entities. As shown in Fig. 6a, there are large-scale short texts in COVID-19 datasets. For short text classification, local features lead to better results than global features. As a result, DC-CNN(Max) always get better performance than DC-CNN (Att) on COVID-19 datasets.

Ablation experimental results of BAAI datasets: Fig. 5b illustrate that the performance of DC-CNN(Max) is slightly poor than DC-CNN, DC-CNN(Att) is highly poor than DC-CNN(Max). As shown in Fig. 6b, only 5% of the news is longer than 80 tokens, and the rest of the news is short. When classifying the large quantity of short news, local features extracted from max-pooling layer are more useful than the long-distance dependency features from attention-pooling layer.

As shown in Fig. 6a and b, both max-pooling layer and attention-pooling layer are considered to capture the local and global information simultaneously.

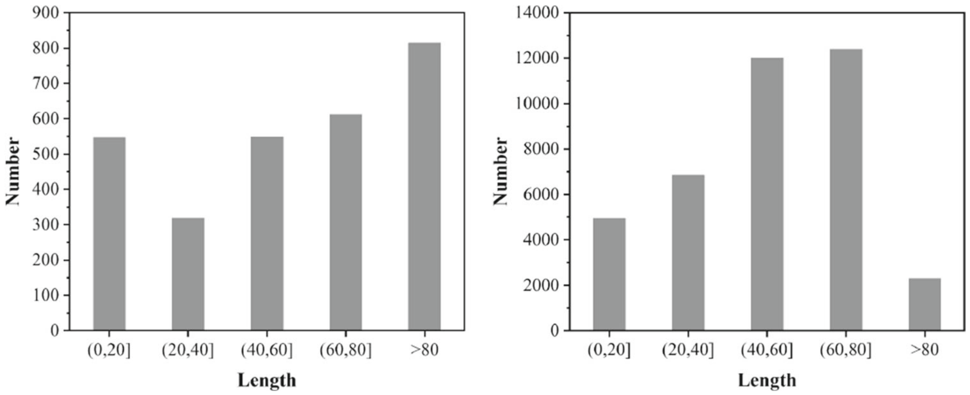


(a) The results on COVID-19 datasets.

(b) The results on BAAI datasets.

Fig. 5 The ablation experimental results

Fig. 6 Data token length distribution



(a) The distribution of text length in COVID-19 dataset. The length of text is calculated after text pre-processing (such as text cleaning and text tokenization).

(b) The distribution of text length in BAAI dataset. The length of text is calculated after text pre-processing (such as text cleaning and text tokenization).

6.3.2 Experimental effects of different lengths data

In this section, multiple groups of experiments were conducted to obtain the performance of DC-CNN in the face of samples of different lengths. According to the length distribution in Fig. 6a and b, COVID-19 and BAAI datasets were split into 5 data sets respectively. In the experiment of this section, we added a new evaluation indicator AUROC (Area Under ROC). DC-CNN is conducted on different datasets, and the experimental results are shown in Tables 8 and 9. We presented the experimental results in the form of a bar chart to help us analyze the impact of different lengths of data. Among them, Table 8 corresponds to Fig. 7a and Table 9 to Fig. 7b.

Experimental results of BAAI datasets: It can be seen from Fig. 7a and b that the accuracy of most experiments will gradually be improved with the increase of data length when the data length is less than 80 tokens. This is because the ability of learning long distance dependency for attention-pooling layer is gradually improved. It indicates that DC-CNN model can cover the shortage of traditional CNN in learning long distance dependency. However, the experimental accuracy will be slightly reduced when the data length exceeds 80. When the data length is too long, the learning effect of the max-pooling layer on local features will be greatly reduced. At this point, the negative effects of max-pooling layer are greater than the positive effects of attention pooling layer, thus, the experimental effect will be slightly reduced.

Experimental results of COVID-19 datasets: In the experiments with COVID-19 dataset, the data effectiveness was significantly reduced when the data length was within the interval of (20,40]. This is due to the small number of samples in this interval. As shown in Fig. 6a, DC-CNN cannot sufficiently learn the semantic information in these data with only about 300 examples in this interval. These experiments show that the attention mechanism can effectively learn long-distance dependency of the relationship.

7 Conclusions

In parallel with the COVID-19 pandemic, fake news has also brought a secondary disaster to society. This paper analyzes the characteristics of fake news in Chinese social media platforms, and proposes a DC-CNN model to detect fake news. DWtext, a dynamic word embedding method,

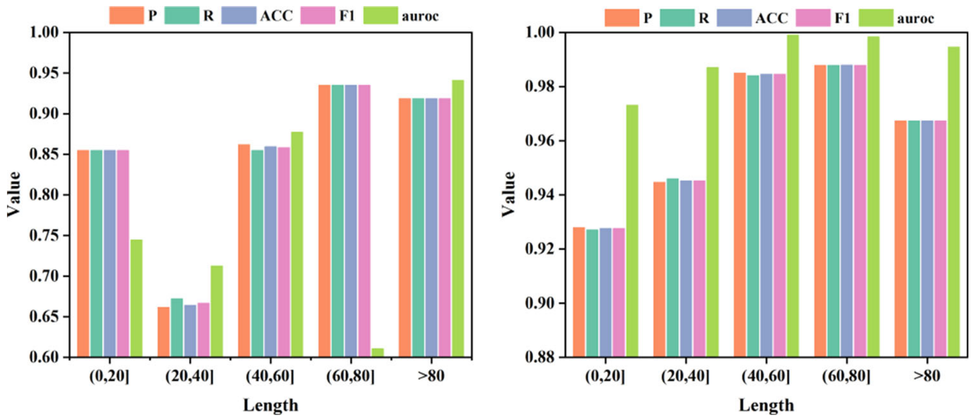
Table 8 Experimental effects of different interval data subsets (COVID-19 datasets)

Section	P	R	ACC	F1	AUROC
(0,20]	0.8545	0.8545	0.8545	0.8545	0.7445
(20,40]	0.6614	0.6718	0.6640	0.6665	0.7126
(40,60]	0.8618	0.8545	0.8590	0.8581	0.8769
(60,80]	0.9349	0.9349	0.9349	0.9349	0.6105
>80	0.9186	0.9186	0.9186	0.9186	0.9409

Table 9 Experimental effects of different interval data subsets (BAAI datasets)

Section	P	R	ACC	F1	AUROC
(0,20]	0.9279	0.927	0.9275	0.9275	0.9731
(20,40]	0.9446	0.9459	0.9452	0.9452	0.9870
(40,60]	0.9850	0.9841	0.9846	0.9846	0.9990
(60,80]	0.9878	0.9878	0.9879	0.9878	0.9983
>80	0.9673	0.9673	0.9673	0.9673	0.9946

Fig. 7 Data token length distribution



(a) Bar chart of experimental effect (COVID-19 datasets).

(b) Bar chart of experimental effect (BAAI datasets).

was proposed in this model to process social media data. It can not only improve the learning ability for non-derived words and valid features, but also reduce noisy data and semantic ambiguity. DC-CNN uses dual-channel pooling layer instead of traditional pooling layer. The Max-pooling layer can extract key information from local features, and the Attention-pooling layer is used to learn long-distance dependency relationships. By combining the learning advantages of the two channels, the problem that CNN is easy to lose the correlation between local features and global features is solved. The experimental results illustrate that the proposed method and model can effectively detect fake news on social media.

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

References

1. Apuke OD, Omar B (2021) Fake news and covid-19: modelling the predictors of fake news sharing among social media users. Telematics Inform 56:101475
2. Bahdanau D, Cho K, Bengio Y (2015) Neural machine translation by jointly learning to align and translate. In: Bengio Y, LeCun Y (eds) 3rd International conference on learning representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings. arXiv:1409.0473
3. Bang Y, Ishii E, Cahyawijaya S, Ji Z, Fung P (2021) Model generalization on COVID-19 fake news detection. In: Chakraborty T, Shu K, Bernard HR, Liu H, Akhtar MS (eds) Combating online hostile posts in regional languages during emergency situation—First international workshop, CONSTRAINT 2021, collocated with AAAI 2021, virtual event, February 8, 2021, Revised selected papers, communications in computer and information science, vol 1402. Springer, pp 128–140. https://doi.org/10.1007/978-3-030-73696-5_13
4. Basiri ME, Nemati S, Abdar M, Cambria E, Acharya UR (2021) Abcdm: an attention-based bidirectional cnn-rnn deep model for sentiment analysis. Futur Gener Comput Syst 115:279–294
5. Cheng Y, Yao L, Xiang G, Zhang G, Tang T, Zhong L (2020) Text sentiment orientation analysis based on multi-channel cnn and bidirectional gru with attention mechanism. IEEE Access 8:134964–134975
6. Feng Y, Cheng Y (2021) Short text sentiment analysis based on multi-channel cnn with multi-head attention mechanism. IEEE Access 9:19854–19863
7. Guo C, Cao J, Zhang X, Shu K, Yu M (2019) Exploiting emotions for fake news detection on social media. arXiv:1903.01728
8. 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
9. Huang Q, Yu J, Wu J, Wang B (2020) Heterogeneous graph attention networks for early detection of rumors on twitter. In: 2020 International joint conference on neural networks (IJCNN). IEEE, pp 1–8
10. Johnson R, Zhang T (2015) Effective use of word order for text categorization with convolutional neural networks. In: Mihalcea R, Chai JY, Sarkar A (eds) NAACL HLT 2015, The 2015 conference of the North American Chapter of the Association for Computational Linguistics: human language technologies, Denver, Colorado, USA, May 31–June 5, 2015. The Association for Computational Linguistics, pp 103–112. https://doi.org/10.3115/v1/n15-1011
11. Johnson R, Zhang T (2017) Deep pyramid convolutional neural networks for text categorization. In: Proceedings of the 55th

annual meeting of the association for computational linguistics (volume 1: long papers), pp 562–570

12. Joulin A, Grave E, Bojanowski P, Mikolov T (2017) Bag of tricks for efficient text classification. In: Lapata M, Blunsom P, Koller A (eds) Proceedings of the 15th conference of the European chapter of the association for computational linguistics, EACL 2017, Valencia, Spain, April 3–7, 2017, volume 2: short papers. Association for Computational Linguistics, pp 427–431. https://doi.org/10.18653/v1/e17-2068
13. Kaliyar RK, Goswami A, Narang P (2021) Fakebert: fake news detection in social media with a bert-based deep learning approach. Multimed Tools Appl 80(8):11765–11788
14. Kowsari K, Jafari Meimandi K, Heidarysafa M, Mendu S, Barnes L, Brown D (2019) Text classification algorithms: a survey. Information 10(4):150
15. Lai S, Xu L, Liu K, Zhao J (2015) Recurrent convolutional neural networks for text classification. In: Twenty-ninth AAAI conference on artificial intelligence
16. Li Q, Zhang Q, Si L (2019) Rumor detection by exploiting user credibility information, attention and multi-task learning. In: Proceedings of the 57th annual meeting of the association for computational linguistics, pp 1173–1179
17. Liu G, Guo J (2019) Bidirectional lstm with attention mechanism and convolutional layer for text classification. Neurocomputing 337:325–338
18. Liu Z, Huang H, Lu C, Lyu S (2020) Multichannel cnn with attention for text classification. arXiv:2006.16174
19. Ma J, Gao W, Mitra P, Kwon S, Jansen BJ, Wong KF, Cha M (2016) Detecting rumors from microblogs with recurrent neural networks
20. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. In: Bengio Y, LeCun Y (eds) 1st International conference on learning representations, ICLR 2013, Scottsdale, Arizona, USA, May 2–4, 2013, workshop track proceedings. arXiv:1301.3781
21. Paka WS, Bansal R, Kaushik A, Sengupta S, Chakraborty T (2021) Cross-sean: a cross-stitch semi-supervised neural attention model for covid-19 fake news detection. Appl Soft Comput 107:107393
22. 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
23. Qi P, Cao J, Yang T, Guo J, Li J (2019) Exploiting multi-domain visual information for fake news detection. In: 2019 IEEE international conference on data mining (ICDM). IEEE, pp 518–527
24. Rakhlin A (2016) Convolutional neural networks for sentence classification. GitHub
25. Rezaeinia SM, Rahmani R, Ghodsi A, Veisi H (2019) Sentiment analysis based on improved pre-trained word embeddings. Expert Syst Appl 117:139–147
26. Rubin VL, Chen Y, Conroy NK (2015) Deception detection for news: three types of fakes. Proc Assoc Inf Sci Technol 52(1):1–4
27. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323(6088):533–536
28. Sabour S, Frosst N, Hinton GE (2017) Dynamic routing between capsules. In: Guyon I, von Luxburg U, Bengio S, Wallach HM, Fergus R, Vishwanathan SVN, Garnett R (eds) Advances in neural information processing systems 30: annual conference on neural information processing systems 2017, December 4–9, 2017. Long Beach, pp 3856–3866. https://proceedings.neurips.cc/paper/2017/hash/2cad8fa47bbef282badbb8de5374b894-Abstract.html
29. Safaei M, Balouchian P, Foroosh H (2018) Ticnn: a hierarchical deep learning framework for still image action recognition using temporal image prediction. In: 2018 25th IEEE international conference on image processing (ICIP). IEEE, pp 3463–3467
30. Shu K, Sliva A, Wang S, Tang J, Liu H (2017) Fake news detection on social media: a data mining perspective. ACM SIGKDD Explor Newsl 19(1):22–36
31. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. In: Guyon I, von Luxburg U, Bengio S, Wallach HM, Fergus R, Vishwanathan SVN, Garnett R (eds) Advances in neural information processing systems 30: annual conference on neural information processing systems 2017, December 4–9, 2017. Long Beach, pp 5998–6008. https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html
32. Vlad GA, Tanase MA, Onose C, Cercel DC (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
33. Vo QH, Nguyen HT, Le B, Nguyen ML (2017) Multi-channel lstm-cnn model for vietnamese sentiment analysis. In: 2017 9th International conference on knowledge and systems engineering (KSE). IEEE, pp 24–29
34. Wang J, Wang Z, Zhang D, Yan J (2017) Combining knowledge with deep convolutional neural networks for short text classification. In: IJCAI, vol 350
35. 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
36. 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
37. Yang C, Zhou X, Zafarani R (2021) CHECKED: Chinese COVID-19 fake news dataset. Soc Netw Anal Min 11(1):58. https://doi.org/10.1007/s13278-021-00766-8
38. Yang J, Li A, Xiao S, Lu W, Gao X (2021) Mtd-net: learning to detect deepfakes images by multi-scale texture difference. IEEE Trans Inf Forens Secur 16:4234–4245
39. Yang J, Xiao S, Li A, Lu W, Gao X, Li Y (2021) Msta-net: forgery detection by generating manipulation trace based on multi-scale self-texture attention. IEEE Trans Circ Syst Video Technol
40. Yoon J, Kim H (2017) Multi-channel lexicon integrated cnn-bilstm models for sentiment analysis. In: Proceedings of the 29th conference on computational linguistics and speech processing (ROCLING 2017), pp 244–253
41. Zhou X, Zafarani R (2018) Fake news: a survey of research, detection methods, and opportunities. CoRR arXiv:1812.00315
42. Zhou X, Zafarani R (2020) A survey of fake news: fundamental theories, detection methods, and opportunities. ACM Comput Surv (CSUR) 53(5):1–40

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

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.

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.

Weijuan Zhang received the Master's degree in computer application technology from University of Jinan in 2009. She is an associate professor at Shandong College of Electronic Technology. Her research interest is natural language processing.

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

Ke Ji was born in 1989. He 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.

Zhenxiang Chen has been a professor of School of Information Science and Engineering at University of Jinan, China since 2015. He received his PhD degree in from Shandong University in 2007. From 2015 to 2016, He had been a postdoctoral fellow in the Department of Electrical and Computer Engineering at Virginia Polytechnic Institute and State University, USA. He research interest is data intensive computing.

Ajith Abraham (Senior Member, IEEE) 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 (H-index of 95 per Google Scholar). He has given more than 100 plenary lectures and conference tutorials (in 20+ countries).

Affiliations

Kun Ma[1] · **Changhao Tang**[1] · **Weijuan Zhang**[2] · **Benkuan Cui**[1] · **Ke Ji**[1] · **Zhenxiang Chen**[1] · **Ajith Abraham**[3]

Changhao Tang
201921200661@mail.ujn.edu.cn

Weijuan Zhang
zhangweijuan@sdcet.cn

Benkuan Cui
202021100414@mail.ujn.edu.cn

Ke Ji
ise_jik@ujn.edu.cn

Zhenxiang Chen
czx@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] Department of Computer and Software Engineering, Shandong College of Electronic Technology, Jinan 250200, China

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