information

MDPI

Article

Dual Co-Attention-Based Multi-Feature Fusion Method for Rumor Detection

Changsong Bing, Yirong Wu, Fangmin Dong, Shouzhi Xu, Xiaodi Liu and Shuifa Sun *

College of Computer and Information Technology, China Three Gorges University, Yichang 443002, China; ChangsongB1ng@gmail.com (C.B.); yirongwu@gmail.com (Y.W.); fmdong@ctgu.edu.cn (F.D.); xsz@ctgu.edu.cn (S.X.); xiaodiliu29@gmail.com (X.L.)
* Correspondence: watersun@ctgu.edu.cn

Abstract: Social media has become more popular these days due to widely used instant messaging. Nevertheless, rumor propagation on social media has become an increasingly important issue. The purpose of this study is to investigate the impact of various features in social media on rumor detection, propose a dual co-attention-based multi-feature fusion method for rumor detection, and explore the detection capability of the proposed method in early rumor detection tasks. The proposed BERT-based Dual Co-attention Neural Network (BDCoNN) method for rumor detection, which uses BERT for word embedding. It and simultaneously integrates features from three sources: publishing user profiles, source tweets, and comments. In the BDCoNN method, user discrete features and identity descriptors in user profiles are extracted using a one-dimensional convolutional neural network (CNN) and TextCNN, respectively. The bidirectional gate recurrent unit network (BiGRU) with a hierarchical attention mechanism is used to learn the hidden layer representation of tweet sequence and comment sequence. A dual collaborative attention mechanism is used to explore the correlation among publishing user profiles, tweet content, and comments. Then the feature vector is fed into classifier to identify the implicit differences between rumor spreaders and non-rumor spreaders. In this study, we conducted several experiments on the Weibo and CED datasets collected from microblog. The results show that the proposed method achieves the state-of-the-art performance compared with baseline methods, which is 5.2% and 5% higher than the dEFEND. The F1 value is increased by 4.4% and 4%, respectively. In addition, this paper conducts research on early rumor detection tasks, which verifies the proposed method detects rumors more quickly and accurately than competitors.

Keywords: social media; rumor detection; co-attention; BERT; early rumor detection

check for updates

Citation: Bing, C.; Wu, Y.; Dong, F.; Xu, S.; Liu, X.; Sun, S. Dual Co-Attention-Based Multi-Feature Fusion Method for Rumor Detection. *Information* **2022**, *13*, 25. https://doi.org/10.3390/info13010025

Academic Editor: Arkaitz Zubiaga

Received: 8 December 2021
Accepted: 5 January 2022
Published: 9 January 2022

1. Introduction

With the rapid development of the mobile Internet, various online social media have become an indispensable part of our daily lives. Twitter, Weibo (Sina Micro-blog), and other social media have the characteristics of instant sharing, easy publishing, and content that do not need to be reviewed. A large number of users are likely to use social media to spread fake information; therefore, internet rumors are becoming a more serious social problem than ever before. In the context of social media, rumors can be defined as unverified speech published by users on social media platforms and spread through the internet [1]. Rumors frequently emerge for certain phenomena and topics, e.g., public health issues and elections [2]. For example, COVID-19 is still a global pandemic issue. To fight against COVID-19, people are encouraged to drink bleach to prevent coronavirus. However, this is a very harmful guidance rumor. For people who decide to take actions based on health advice found online, it is often difficult for them to understand potential risks from disinformation [3]. To solve the problem of the flood of rumors on social media, social platform companies and government agencies mostly use conventional manual detection methods, that is, they rely on human experts to detect rumors [4,5]; however, the labor cost of these conventional methods is high.

With the development of automatic rumor detection research, more researchers are focusing on methods based on machine learning with feature engineering. Gao et al. [6] regarded rumor detection as a binary classification problem with supervised learning; Zhang et al. [7] used a variety of potential features such as opinions and emotional polarity in the comments; Zhao et al. [8] used regular expressions to match the questioning information in the text; Ma et al. [9] used dynamic time series models and support vector machine (SVM) to detect rumors. However, feature engineering needs to consume a great deal of manpower, material resources, and time. Additionally, those methods mostly rely on the information collected at the later stage of rumor spread, such as emotional polarity, questioning, and refutation, which results in a lag for rumor detection.

In recent years, deep learning methods have been widely used to obtain feature representations of text contents of rumors instead of conventional machine learning methods. For example, Ma et al. [10] used recurrent neural networks (RNNs) to process text sequence data for rumor detection. Natali et al. [11] used Singular Value Decomposition (SVD) to acquire group behavior of participating users, and modeled text contents using Long Short-Term Memory (LSTM). Tu et al. [12] proposed a novel rumor detection framework with joint text and propagation structure representation learning. Yuan et al. [13] improved CNN-based model and proposed a dilated-block-based convolutional network to detect rumor. Xu et al. [14] proposed a rumor detection method that utilizes both GCN and GNN to update text-level and word-level features. Although these rumor detection methods have achieved good performance, rumor detection is a comprehensive task involving features from different sources, such as user profiles, source tweets, and associated comments, which would be benefit for rumor detection. However, it is challenging to make full use of these features to timely and effectively recognize and curb the rumors before they break out.

Considering existing problems in current rumor detection methods, such as unsatisfactory representation of potential text features, inadequate use of information at the initial stage of rumor spreading, and the lag in rumor detection, this paper proposes a BERT-based Dual Co-attention Neural Network (BDCoNN) integrating features from publishing user profiles, tweets, and comments. The contributions of our study are as follows:

1. The proposed method acquires user characteristics from user profiles, and extracts key features from source tweets and comments using BERT and hierarchical attention mechanisms.
2. A dual collaborative attention mechanism is proposed to explore the correlation between publishing user profiles and tweet contents, and between tweet contents and associated comments.
3. The experimental results on Weibo, CED, and other datasets show that the proposed method significantly improves the performance of rumor detection, and performs well in early rumor detection tasks.

The remainder of this paper is organized as follows: In Section 2, relevant methods for rumor detection on social media are reviewed; in Section 3 the proposed BDCoNN method is systematically elaborated; in Section 4, the experiment and results are presented; in Section 5, the study is summarized and future work is proposed.

2. Related Works

In recent years, rumor detection on social media has attracted wide attention. Rumor detection methods can be divided into the following categories.

2.1. Content-Based Methods

Zhang et al. [7] developed a method to detect rumors using features related to emotional polarity and social influence. Zhao [8] classified rumor posts by matching query phrases with regular expression. Ma et al. [10] acquired relevant semantic features of text sequences using an RNN, and alleviated the problems of gradient disappearance and gradient explosion using LSTM and a gated recurrent unit (GRU). Chen et al. [15] combined the attention mechanism with an RNN to learn hidden representations by capturing the long-distance dependencies of tweet sequences. Ma et al. [16] proposed a generative adversarial network (GAN) model based on RNN for rumor detection, which simulated irrelevant feature changes of rumors in a time sequence using a generator, and used adversarial training to improve the capability of the discriminator.

2.2. User-Based Methods

Yang et al. [17] developed an SVM classifier for rumor detection using features from user profiles, such as geographical location and authentication information of users on the Sina Weibo platform. Shu et al. [18] considered information such as user political complexion and personal profile pictures for rumor detection, and verified that observable differences exist in user characteristics between rumor spreaders and non-rumor spreaders. Liu et al. [19] used CNN and RNN to capture the global and local changes of user characteristics along the propagation path to detect fake news. Castillo et al. [20] designed statistical features based on Twitter contents, such as emotion score and text length, and used the changes of some user characteristics in the message life cycle to classify and detect rumors with the decision tree algorithm.

2.3. Hybrid Methods

Natali et al. [11] modeled news contents using LSTM, and acquired the group behavior of participating users using SVD algorithm. Jin et al. [21] used the pre-trained VGG-19 model to extract visual features, which were combined with text and social context features to classify rumors. Qi et al. [22] proposed a multi-domain visual rumor detection model, which combined visual features in frequency domain and pixel domain to classify and detect rumors from both physical and semantic levels. Shu et al. [23] proposed a fake news detection model that learned the correlation between long twitter text and comments, and explored the interpretability of the false news detection model. Zhou et al. [24] designed an early rumor detection model, including a rumor detection module and a checkpoint module. The rumor detection module is composed of a word embedding layer, a maximum pooling layer, and a gated neural unit.

2.4. Propagation-Based Methods

Bian et al. [25] introduced a graph convolutional network (GCN) to explore the mechanisms of top-down and bottom-up rumor propagation. Ma et al. [26] proposed a tree-based RNN based on the non-sequential propagation mechanisms of tweets to classify and detect rumors. Lu et al. [27] used GCN to capture the information of direct or indirect neighbor nodes to simulate potential interaction of users. At the same time, they used a dual collaboration attention mechanism to capture user interaction and the correlation between original text and user profiles to detect fake news. Although the above research has achieved good performance for rumor detection, some problems still exist. For example, word2vec and Glove are widely used in text semantic representation but they cannot be used to solve the polysemy problem. Additionally, in some studies, researchers have typically focused on one or two aspects of features. Methods such as those in [28] rely heavily on manually acquired features, and methods such as those in [23] ignore user profile features. Moreover, methods such as those in [25] rely on the propagation and diffusion of GCN node information, and a lag exists in rumor detection.

3. Bdconn Method for Rumor Detection

The proposed BDCoNN method uses features from user profiles, original tweets, and corresponding comments to detect rumors. The BDCoNN consists of four modules:

(1) User feature module. This module is used to quantify user information and extract its vector representation.

(2) Source text encoding module. This module is used to generate the vector representation of the publisher's source text.
(3) Comment feature extraction module. This module is used to extract the sentence vector representation from the user's comments.
(4) Dual co-attention module. This module is used to extract the correlation between user profiles and source tweets, and the correlation between source tweets and comments.

The overall network model architecture is shown in Figure 1.

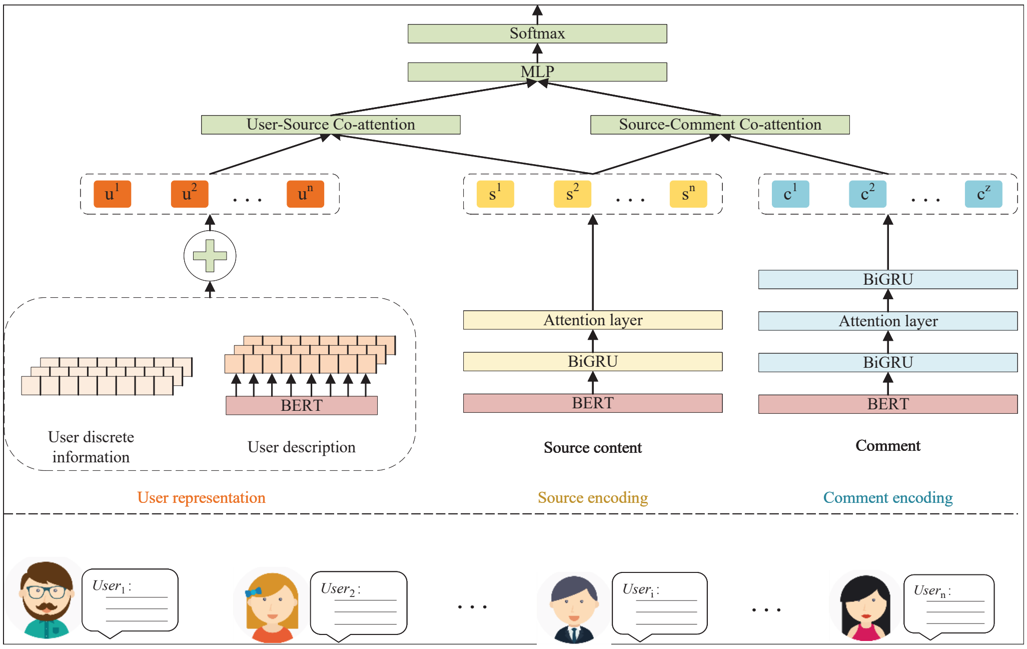

Figure 1. Overall network model architecture.

3.1. User Feature Module

Different from traditional news media, such as newspapers, radio, and television, online social media has unique characteristics, that is, users are the mainstay on social media, and each user may be the publisher of rumors; therefore, research on user information is very important. Rumor disseminators and non-rumor information disseminators have different characteristics [29]. According to Liu's work [28], to evade relevant liability, some rumor mongers use certified accounts to release rumors, whereas normal users use certified accounts to improve their popularity. Additionally, rumor bloggers who profit from We Media hope to gain more fans, and simultaneously, they publicize some fake business information in their personal descriptions. Moreover, social media users may also be social robots that maliciously spread information. In summary, in social media, user profiles can reflect user behavior characteristics and user-influence on the public. Table 1 shows several examples of user information in this study. User information or a user profile is mostly composed of user discrete information and user descriptions.

3.1.1. User Discrete Information

For gender, account authentication, and other discrete values, one-hot coding approach is adopted. For other discrete information, such as the number of followers, because a user who has a high influence on the public has more followers than ordinary users, maximum and minimum normalization is adopted to eliminate the influence caused by extreme values. The standard normalization formula is described as follows:

$$x' = \begin{cases} \frac{logx - logx_{min}}{logx_{max} - logx_{min}}, & x > 0 \\ 0, & x = 0 \end{cases} \tag{1}$$

where x represents the value before normalization, $x^{'}$ represents the normalized standard value, x_{max} and x_{min} represent the maximum and minimum values in the dataset, respectively. Because CNN focuses on local information, one-dimensional convolution is used to process the input user information. The width of the one-dimensional convolution kernel is set to the width of the user information matrix to extract overall features and learn the sequence correlation of user profile features. For the input user information matrix $X_i = \{ui_1, ui_2, ..., ui_n\}$, the user feature formula generated by one-dimensional convolution is expressed as as eigenvector V_{ui}:

$$V_{ui} = ReLU(W_{ui} \cdot X_u + b) \tag{2}$$

where $ReLU$ is the activation function, W_{ui} is the weight matrix, b is the bias term.

Table 1. Examples of social media user information.

User Name	User Discrete Information	User Description
Sogou input method	bi_followers_count:368 friends_count: 693 followers_count:2985442 statuses_count:6980 favourites_count:490 comments_count:45	[Sogou input method smart version 2.0] lick on the light bulb or the number “0” to trigger, the software, film and television music, and the website will be available immediately after input; Chameleon, perceives the input environment, changes the color, and provides more accurate input candidates. download link: xxx[Business Cooperation] QQ: xxx
Guangzhou part-time full-time official website	bi_followers_count:435 friends_count: 1516 followers_count:636767 statuses_count:8285 favourites_count:13 comments_count:72	For companies to post recruitment information or part-time full-time recruitment, please contact QQ: xxx, and release the latest and most complete Guangzhou part-time full-time internship information every day. All the information released is free of card charges [please add QQ directly, not private messages]

bi_followers_count: the number of bi-directional followers.

3.1.2. User Descriptions

For the text content in the user identity descriptions, the BERT model is used in the word embedding layer. After user descriptions pass through the TextCNN, the results are concatenated with the discrete user feature vector extracted by the one-dimensional CNN to obtain the overall representation of user information.

The BERT model based on a self-attention mechanism uses a bidirectional transformer structure in the pre-training phase of language models [30]. Compared with word2vec, it overcomes the unidirectional limitations. It is a very effective algorithm to obtain text feature representation. This module uses the BERT-base-Chinese model with 768-dimension for text feature extraction, and uses the output of the penultimate layer of the encoder as the word vector representation. The user description text $U_d = \{ud_1, ud_2, ..., ud_n\}$ is a sequence of text with fixed length T_u. If text is too long, it is truncated; if text is insufficient, some text is padded. Then, after token embedding, position embedding and segment embedding are added to obtain vector representation of the user descriptions $E_u \in R^{T_u \times H}$, where H is the feature vector dimension and

$$E_u = E_{Token} + E_{Segment} + E_{Position} \tag{3}$$

A user description is different from paragraphs or articles, and its syntax and semantics are difficult to understand. Our user feature module uses TextCNN [31] to extract features from user descriptions since it has a strong capability of extracting local features of short text. For an input matrix of size $T_u \times H$, the kernel of $W_{ud} \in R^{h \times H}$ and $X_{i:i+H-1}$ are used to perform the convolution operation, where h is the filter size and is set to 2, 3, and 4. $X_{i:i+H-1}$ is obtained using the input matrix from row i to row $i + H - 1$. Then the eigenvector V_{ud} is obtained as

$$V_{ud} = f(W_{ud} \cdot X_{i:i+H-1} + b) \tag{4}$$

where b is the bias term and f is a nonlinear function. Finally, the eigenvectors are obtained from the user discrete information V_{ui} and user description V_{ud} are concatenated to obtain the overall vector representation of user profile features $U \in R^{T_u \times 2H}$,

$$U = V_{ui} \bigoplus V_{ud} \tag{5}$$

3.2. Source Text Encoding Module and Comment Feature Extraction Module

Compared with regular text, rumors contain more deliberate and inflammatory expressions. Simultaneously, there are doubts about these expression and refutations in comments. It is beneficial to improve the accuracy of rumor detection by capturing the correlation between these key items of semantic information. The first step is to preprocess the data that includes source tweets and comments, and remove redundant information, such as URL links and the meaningless expression “repost” in comments. Then, according to the characteristics of real social media microblogs with limited content length, a hierarchical attention mechanism is introduced to learn the weights of key features from word level and sentence level, respectively

Source tweets are modeled as follows. Source tweets are considered as a set of short text $I = \{s_1, s_2, ..., s_n\}$ as input. A fixed length T_s is set for tweet sentence s_i, where $s_i = \{w_1, w_2, ..., w_{T_s}\}$ composed of T_s words. BERT model is used to extract the word vector representation $E_s = \{e_1, e_2, ..., e_{T_s}\}$, where $e_i \in R^{T_s \times H}$. GRU is used to address the issue of gradient disappearance for a long sequence. Word-level timing information and semantic information in microblog data are mined from two directions using BiGRU as follows,

$$h_t = BiGRU(e_t) \tag{6}$$

where h_t is a hidden vector that contains forward and backward information. Then an attention mechanism is introduced to learn the weight of key features in the hidden vector. The calculation process is given as follows, and the attention layer is shown in Figure 2.

$$y_i = tanh(W_y \cdot h_i + b) \tag{7}$$

$$\alpha_i = \frac{exp(y_i W_\alpha)}{\sum_{j=1}^{n} exp(y_j W\alpha)} \tag{8}$$

$$s^i = \sum_{1}^{T_s} \alpha_i h_i \tag{9}$$

The output h_t of BiGRU is weighted. Different weights α_i are specified so that the model can focus on the key sequence information. The vector representation y_i of the hidden sequence h_t is obtained through the fully connected layer, and then the weight α_i is calculated according to y_i. Finally, the source text vector representing $s^i$is obtained, and the final source text is represented as $S \in R_s^T \times 2H$.

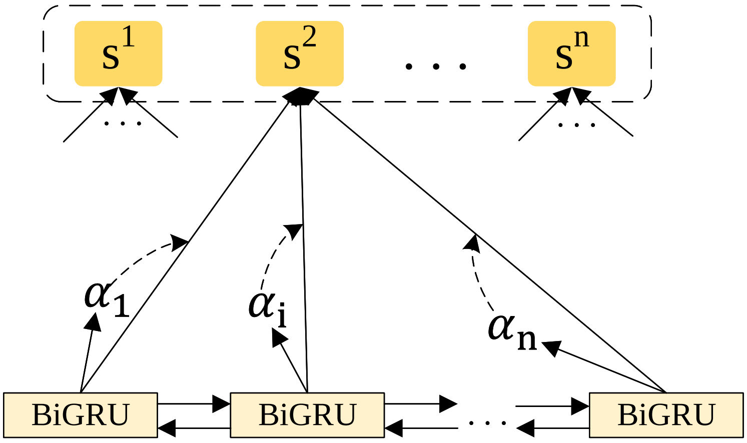

Figure 2. Attention layer.

In this study, comments are modeled as follows. Rumor information is highly controversial, and it typically induces heated discussions to expand social influence to achieve the profit purpose. People express their emotions and opinions on the rumor information, such as words of doubt and correction that appear frequently in the microblog of the rumor, for example, “Is it true?” and “The news is not true”. This kind of information is a very useful for rumor identification. The comments of microblog events are modeled as long text $R = \{c_1, c_2, ..., c_z\}$, that is, z comments corresponding to the source text are selected to form a long text, where $c_i = \{w_1, w_2, ..., w_{T_c}\}$, T_c is the fixed length of the selected comments. The final input for the module is a long text $L = \{R_1, R_2, ..., R_n\}$. The modeling process of word-level vector representation for comments is the same as that for source text. The sentence representation can be described as $v^i = \sum_1^{T_c} \beta_i h_i$, which is similar to the feature representation of the level in the processing of long text [32]. BiGRU is used as the encoder to extract the sentence context information from two directions:

$$c^i = BiGRU(v^i) \tag{10}$$

3.3. Dual Co-Attention Module

Previous studies have indicated that a few influential and badly motivated users play a leading role in spreading rumors [29]. The degree of repercussions they arouse after publishing rumors is related to their user behavior characteristics and influence on the public. A user feature module is proposed to extract the representation of user behavior and influence to simulate user portraits, to a certain extent. Simultaneously, for rumors released by the user and the user itself, leveraging the idea of the correlation between image and text in the VQA task [33], and the idea of its expansion [27], this paper proposed a collaborative attention mechanism that captures the association between source tweet and user profile. Similarly, a collaborative attention mechanism was proposed to capture the association between the source text and the comments. Both attention mechanisms are used to obtain the hidden representation of the entire event for rumor classification. The model details are shown in Figure 3.

To obtain the collaborative attention between the source tweet and user profiles, the correlation matrix P is first calculated as a new feature:

$$P = tanh(U^T W_x S) \tag{11}$$

where $W_x \in R^{T_u \times T_s}$ is a weight matrix. Using the correlation matrix $P \in R^{(2H \times 2H)}$, the attention distribution of U and S can be predicted.

$$H^u = tanh(W_u U + (W_s S)P^T) H^s = tanh(W_s S + (W_u U)P) \tag{12}$$

where $W_u \in R^{k \times T_u}$ and $W_s \in R^{k \times T_s}$ are weight matrices. The correlation matrix P converts the user feature attention space to the source text attention space, and P^T performs the opposite conversion. The attention weights a^u and a^s of the source text and user features are calculated by a $softmax$ function, described as follows,

$$a^u = softmax(w_{hu}^T H^u) a^s = softmax(w_{hs}^T H^s) \tag{13}$$

where a^u and a^s are the attention probabilities of each user profile and source text, respectively. w_{hu} and $w_{hs} \in R^{1 \times k}$ are weight matrices. Finally, the attention vectors of user profile features $\hat{u}$ and source text $\hat{s}$ can be obtained as a weighted summation of the output attention weights, described as follows:

$$\hat{u} = \sum_{i=1}^{T_u} a_i^u u^i, \hat{s} = \sum_{j=1}^{T_s} a_j^s s^j \tag{14}$$

where $\hat{u} \in R^{1 \times 2H}$ and $\hat{s} \in R^{1 \times 2H}$ are user profile feature vectors and source text feature vectors, respectively, which are learned using co-attention mechanisms. The collaborative attention mechanism between the source text $S \in R^{T_s \times 2H}$ and the user comment $C \in R^{T_c \times 2H}$ is processed in the same manner as the collaborative attention mechanism between the user profiles and the source text, the details are omitted here.

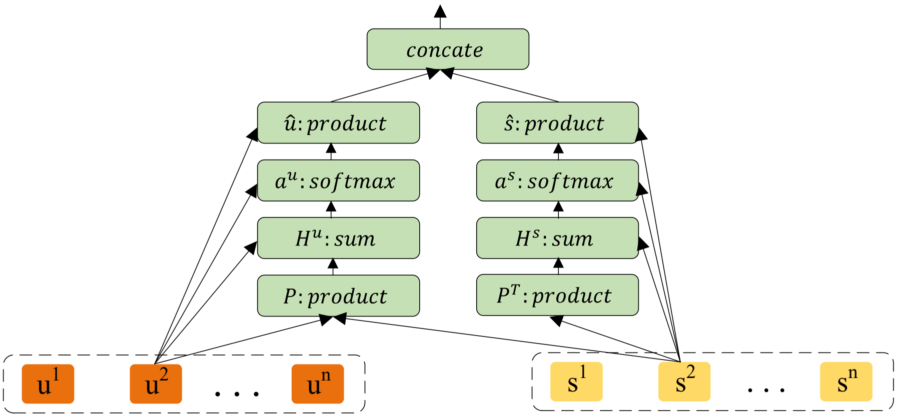

Figure 3. User-source co-attention module.

3.4. Prediction

The dual cooperative attention mechanism is used to obtain the user profile features $\hat{u}$ source text features $\hat{s}$, and comment featurs $\hat{c}$ for rumor detection. Assume $\gamma = [[\hat{u}, \hat{s}], [\hat{s}, \hat{c}]]$, then

$$\hat{y} = softmax(MLP(\gamma)) \tag{15}$$

MLP is used to compress the dimension of the input feature vector to the same feature dimension as the number of categories, and then the feature vectors are classified using $softmax$. Cross entropy is used as the $loss$ function:

$$loss = -ylog(\hat{y}_1) - (1 - y)log(1 - \hat{y}_0) \tag{16}$$

4. Experiment Results and Analysis

4.1. Dataset

Publicly available Weibo dataset and CED dataset were used in the experiment. The Weibo dataset is a rumor event dataset published by Ma et al. [9], who used a microblog API to collect a large amount of rumor events from the Sina microblog community management center, and used crawlers to collect and extract non-rumor events. The CED dataset was proposed by Song et al. [34]. Similar to the Weibo dataset, it also contains user information, original microblog, and associated comments. The statistics of the datasets are shown in Table 2.

Table 2. The statistics of the dataset.

Statistics	Weibo	CED
User	2,746,818	1,278,567
Posts	3,805,656	1,392,561
Events	4664	3387
Rumors	2313	1538
Non-Rumors	2351	1849

4.2. Baseline Methods

The following baseline methods were selected for comparison with the proposed method:

(1) TextCNN [31]: TextCNN uses character-level word2vec to convert microblog events that are composed of source text and comments into vectors. A CNN is utilized to process these vectors, in which multiple convolution filters are used to capture features with different granularities.

(2) HAN [32]: HAN uses word-level word2vec to map Weibo events into vector representations. Its hierarchical attention network structure assigns different attentions to words and sentences. A "word-sentence-event" hierarchical structure is used to represent an event.

(3) GRU-2 [9]: GRU-2 arranges the posts in each microblog in chronological order. Using the time interval generation algorithm, the posts are roughly divided into N parts with the same interval. A two-layer GRU network is used, and its input is the TF-IDF values of the text in the current time interval.

(4) dEFEND [23]: dEFEND is a collaborative attention model for learning the association between long news and its comments. dEFEND captures the top-K important sentences in long news through a hierarchical attention mechanism. These important news sentences and their associated comments are used to generate a representation of the microblog.

(5) TextGCN [35]: TextGCN constructs the corpus into a graph, uses one-hot encoding to process words and documents, and uses TF-IDF values as the weights of the document-word edge to capture global word co-occurrence information, and the relationship between documents and words.

(6) BERT [30]: BERT is a pre-training language model based on a transformer, and uses MLM to generate a deep bidirectional language representation, which performs well in multiple NLP tasks. In its implementation, feature vectors extracted by BERT are input to the fully connected layer to categorize microblog rumor events.

4.3. Experimental Evaluation Index

In this study, to evaluate the performance of the methods for rumor detection, four evaluation metrics were selected: Accuracy, Precision, Recall, and F1 value.

(1) Accuracy: It is defined as the proportion of the correctly predicted events among all events.

(2) Precision: It represents the proportion of actual rumors among all events predicted to be rumors.

(3) Recall: It indicates the proportion of correctly predicted rumors in all actual rumors.

(4) F1: It is a weighted harmonic average of the Precision and Recall, which is a comprehensive consideration of Precision and Recall.

4.4. Experimental Setup

The method is developed on the PyTorch platform in Linux x86_64, GTX2080Ti system. The dataset is divided into three parts: training set, verification set, and test set according to the ratio of 80%, 10%, and 10%. The pre-trained BERT model with 12 layers and 768 dimensions is utilized to extract vectors, and the output of the penultimate layer in BERT is taken as the text representations. The filter size of CNN in user feature module is set to 2, 3, and 4. In the text encoding module and comment feature extraction module, the input size and output size of bi-gru are set to 768. The Adam algorithm is used to optimize the parameters of the networks. The learning rate is set to 1×10^{-5}, and the learning rate decreased adaptively. Epoch is set to 10 and if the loss does not decrease within 10 epochs, the training process is stopped immediately.

4.5. Experimental Results and Analysis

4.5.1. Rumor Detection Performance

To verify the effectiveness of proposed method, this paper compared the rumor detection performance between our method and baseline methods. Comparison results are shown in Table 3. The category (class) was divided into R (rumor) and N (non-rumor). The accuracies of proposed BDCoNN on the Weibo and CED datasets are 0.957 and 0.946, respectively, which are 3.9% and 1.9% higher than BERT, and 5.2% and 5% higher than the dEFEND method. The proposed method achieves better rumor detection performance than baseline methods. Additionally, the experimental results in Table 3 show that, compared with the best baseline method, the proposed method improves the F1 value of rumor detection by 4% and 2% on the Weibo and CED datasets, respectively. Moreover, the results show that HAN performs poorly for rumor detection using source text and comments. One of the possible reasons is that the structure of Chinese sentences is different from that of naturally separated English sentences, which may impact the performance of data preprocessing. For Chinese sentences, Jieba should be used to cut sentences, identify stop words, and remove some special symbols, which inevitably causes a lack of information and impacts the classification performance. Other traditional word segmentation methods also limit the performance of HAN. The dEFEND method distinguishes the source tweets from the comments, and learns the sequence vector representation separately. However, it uses a preprocessing method, which is similar to that in HAN. A great amount of effective information is lost. The TextCNN method uses character-level vectors, and the extracted feature information is relatively sufficient. However, special symbols are included, which impact detection performance. In this method, CNN is used to capture local information, resulting in some limitations when detect rumors on social media. It is necessary to consider time information to improve rumor detection performance.

The results also show that GRU-2 is better than TextCNN and HAN on detection accuracy. GRU-2 considers the time information in the microblog, and performs modeling and analysis on text sequence data to obtain the hidden features of the rumor context, which changes over time. However, it does not use a large amount of user information in real social media. TextGCN converts the text into more general unstructured graph, in which convolution kernel acts on all nodes of the entire graph. The overfitting issue can be effectively avoided, and it achieves the same performance as GRU-2. The BERT model benefits from a powerful self-attention mechanism and MLM method, which enable it to capture two-directional features of text. It achieves a high performance in classification but it does not consider the relevance of rumors in social media to users. The proposed BDCoNN method makes full use of various information, such as user information, source text, and comments. It collaboratively learns the joint representation of the three types of information, which solves the problem of low generalization for rumor detection on social media. The following ablation experiments are conducted to prove this point.

Table 3. Comparison of experimental results table.

Method	Class	Weibo				CED			
		Accuracy	Precison	Recall	F1	Accuracy	Precison	Recall	F1
HAN [32]	R	0.845	0.834	0.852	0.847	0.852	0.853	0.865	0.859
	N		0.860	0.843	0.851		0.885	0.842	0.863
TextCNN [31]	R	0.875	0.883	0.854	0.868	0.871	0.859	0.867	0.863
	N		0.867	0.893	0.880		0.888	0.881	0.885
GRU-2 [9]	R	0.910	0.876	0.916	0.898	0.906	0.890	0.916	0.898
	N		0.952	0.864	0.906		0.913	0.898	0.905
dEFEND [23]	R	0.905	0.863	0.923	0.892	0.896	0.873	0.909	0.891
	N		0.940	0.891	0.915		0.912	0.902	0.907
TextGCN [35]	R	0.881	**0.960**	0.842	0.897	0.899	0.896	0.892	0.894
	N		0.790	0.944	0.860		0.896	0.892	0.904
BERT [30]	R	0.918	0.954	0.881	0.916	0.927	0.895	0.951	0.926
	N		0.887	**0.956**	0.920		0.960	0.897	0.928
BDCoNN	R	**0.957**	0.943	**0.969**	**0.956**	**0.946**	**0.916**	**0.978**	**0.946**
	N		**0.970**	0.947	**0.959**		**0.978**	**0.918**	**0.947**

Values in bold represent the best result in each category among all methods.

4.5.2. Ablation Experiments

A series of ablation experiments were conducted to illustrate the influence of each module in the proposed method on the rumor detection task. The experimental results are shown in Figure 4. BDCoNN denotes a complete model that uses all modules. The ablation experiments are organized as follows,

(1) BDCoNN-User: In this experiment, the user feature module was removed, and only the source text and its corresponding comments were considered. The vector representation through the dual co-attention module is obtained, then feed into MLP and softmax layer for rumor prediction.

(2) BDCoNN-Source-Dual: In this experiment, the source text encoding module and the dual co-attention module were removed. The hidden vector extracted from the user feature module was concatenated with the comment hidden vector, which was input into the fully connected layer and softmax layer for rumor detection.

(3) BDCoNN-Comment: In this experiment, the comment feature extraction module was removed, and only the user profile information and the source text were encoded. The correlation between the user feature and the source text feature was learned using the co-attention mechanism, and the results were feed into MLP and softmax layer for classification.

(4) BDCoNN-Dual-Co: In this experiment, the dual co-attention module was removed. The feature vectors of user profiles, source text, and comments were concatenated, and then feed into MLP layer and softmax layer for classification.

Figure 4 shows the ablation experiment results on two datasets. The results show that after user features are removed, the model is less affected on the CED dataset than that on the Weibo dataset. One of the reasons is that approximately 3% of user data is missing in the CED dataset in the data preprocessing stage, which results in an insufficient contribution to model performance. When the source text encoding module and the dual co-attention module are eliminated, the performance of the model decreases, with a decrease of 4.03% and 6.48% on the Weibo and CED datasets, respectively, which clearly demonstrates the importance of source text for rumor detection. After the dual co-attention module is removed, the reduction in the model performance is the second only to that of BDCoNN-Source-Dual, which indicates that the dual co-attention mechanism makes distance-related semantic information close to each other. It can be used to distinguish the rumor and non-rumor groups, thus improving performance. The experiment on the

BDCoNN-Comment module also indicates that the characteristics of comments are of great significance for rumor detection.

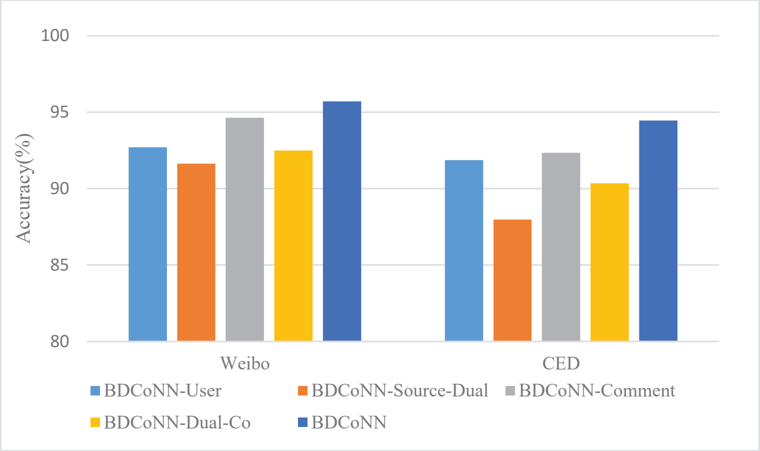

Figure 4. Results of the ablation experiments on the Weibo and CED datasets.

4.5.3. Early Rumor Detection

The best scenario for rumor detection is to identify a rumor before the outbreak and reduce its harm. The goal of early rumor detection is to identify a rumor at the initial stage of rumor spreading. To evaluate the effectiveness of BDCoNN in early rumor detection tasks, a new test set is developed, including publishing user profiles, source tweets, and the comments provided by the first 30 people. The experimental results are shown in Figures 5 and 6.

Figures 5 and 6 show that the proposed BDCoNN is significantly better than the baseline methods for early rumor detection. Even when there are no comments, the accuracy of rumor detection still reaches 94.64% and 92.35% on the Weibo and CED datasets, respectively. Compared with dEFEND, GRU-2, and HAN, the proposed method relies less on microblog information within the time range. This demonstrates that BDCoNN alleviates the reliance on rich microblog information for detection, whereas the baseline methods require such information. BDCoNN effectively solves the lag problem in detection, and achieves the highest performance in the shortest time, which is of great significance for the early rumor detection.

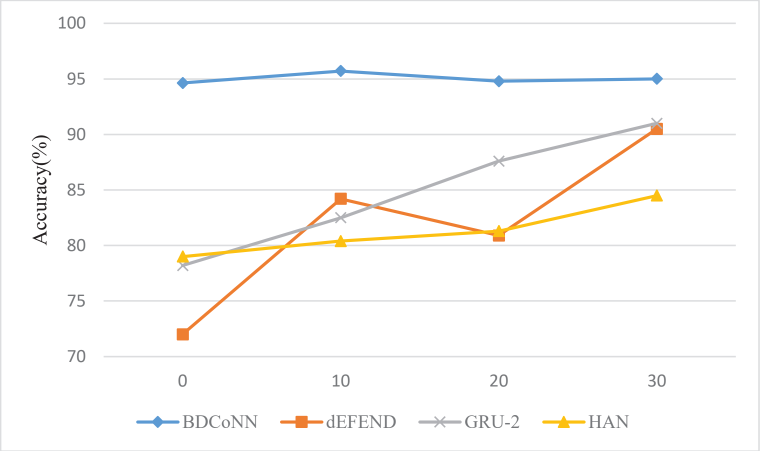

Figure 5. Accuracy of early rumor detection on the Weibo dataset.

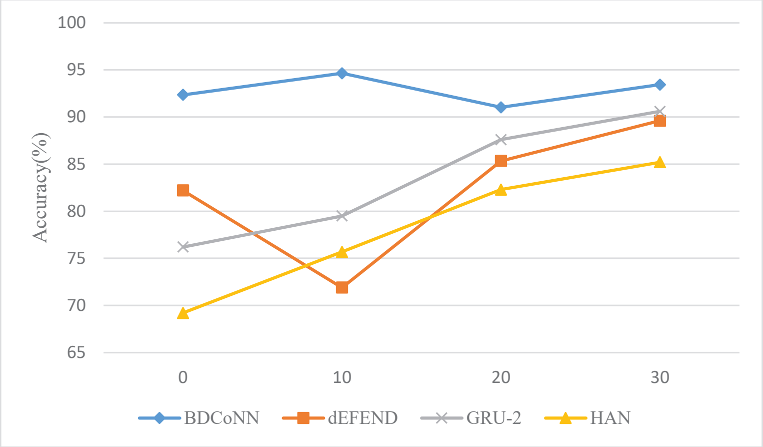

Figure 6. Accuracy of CED's early rumor detection.

5. Conclusions and Future Work

In this study, a dual co-attention based multi-feature fusion method for rumor detection is proposed. This method is used to detect rumors on social media by modeling discrete and continuous user profile data to obtain the hidden layer sequence representation. Additionally, rich semantic information in source text and comments is captured using BERT. Moreover, BiGRU is used to learn the context-related information in the sequence, and the dual co-attention mechanism is finally introduced to obtain the relevant hidden layer sequence representation of the user profiles, source text, and comments for rumor detection. We investigated the influence of various features in social media on the detection of rumors. The ablation study further demonstrates that each module in our model is indispensable for rumor detection. On the real-world datasets of Weibo and CED, the proposed rumor detection method outperformed the baseline models. In early rumor detection tasks, the proposed method performs better than state-of-the-art methods. There are also several limitations in our work. In some more professional topics such as finance, public health issues, etc., the accuracy of the method may be affected when the method is used to detect rumors in some professional areas such as finance, public health, etc. Additionally, we trained the neural network on the Chinese social media datasets only, which results in low generalization performance. Further research should explore contrastive learning of unsupervised or self-supervised learning for rumor detection, and consider how to improve the model generalizability for datasets with a small scale and inconsistent data structure.

Author Contributions: Conceptualization, C.B.; methodology, C.B.; software, C.B.; validation, C.B.; formal analysis, C.B.; investigation, S.S. and S.X.; resources, C.B. and X.L.; data curation, C.B.; writing—original draft preparation, C.B.; writing—review and editing, Y.W. and S.S.; visualization, C.B.; supervision, S.S. and Y.W.; project administration, F.D.; funding acquisition, F.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (Grant No. U1703261).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data included in this study are available upon request by contact with the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bondielli, A.; Marcelloni, F. A survey on fake news and rumour detection techniques. *Inf. Sci.* **2019**, *497*, 38–55. [CrossRef]
2. Dharawat, A.; Lourentzou, I.; Morales, A.; Zhai, C. Drink bleach or do what now? covid-hera: A dataset for risk-informed health decision making in the presence of covid19 misinformation. *arXiv* **2020**, arXiv:2010.08743.
3. Morales, A.; Narang, K.; Sundaram, H.; Zhai, C. CrowdQM: Learning aspect-level user reliability and comment trustworthiness in discussion forums. *Adv. Knowl. Discov. Data Min.* **2020**, *12084*, 592.
4. Zhou, X.; Zafarani, R. A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities. *ACM Comput. Surv.* **2020**, *53*, 109. [CrossRef]
5. Zubiaga, A.; Aker, A.; Bontcheva, K.; Liakata, M.; Procter, R. Detection and Resolution of Rumours in Social Media: A Survey. *ACM Comput. Surv.* **2018**, *51*, 32. [CrossRef]
6. Gao, Y.; Liang, G.; Jiang, F.; Xu, C.; Yang, J.; Chen, J.; Wang, H. Social Network Rumor Detection: A Survey. *Acta Electron. Sin.* **2020**, *48*, 1421–1435.
7. Zhang, Q.; Zhang, S.; Dong, J.; Xiong, J.; Cheng, X. Automatic Detection of Rumor on Social Network. In *Natural Language Processing and Chinese Computing*; Li, J., Ji, H., Zhao, D., Feng, Y., Eds.; Springer International Publishing: Cham, Switzerland, 2015; pp. 113–122.
8. Zhao, Z.; Resnick, P.; Mei, Q. Enquiring Minds: Early Detection of Rumors in Social Media from Enquiry Posts. In Proceedings of the WWW '15: 24th International Conference on World Wide Web, Florence, Italy, 18–22 May 2015; International World Wide Web Conferences Steering Committee: Geneva, Switzerland, 2015; pp. 1395–1405. [CrossRef]
9. Ma, J.; Gao, W.; Wei, Z.; Lu, Y.; Wong, K.F. Detect Rumors Using Time Series of Social Context Information on Microblogging Websites. In Proceedings of the CIKM '15: 24th ACM International on Conference on Information and Knowledge Management, Melbourne, Australia, 18–23 October 2015; Association for Computing Machinery: New York, NY, USA, 2015; pp. 1751–1754. [CrossRef]
10. Ma, J.; Gao, W.; MItra, P.; Kwon, S.; Jansen, B.J.; Wong, K.F.; Cha, M. Detecting rumors from microblogs with recurrent neural networks. In Proceedings of the IJCAI '16: 25th International Joint Conference on Artificial Intelligence, New York, NY, USA, 9–15 July 2016; pp. 3818–3824.
11. Ruchansky, N.; Seo, S.; Liu, Y. CSI: A Hybrid Deep Model for Fake News Detection. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, Singapore, 6–10 November 2017; Association for Computing Machinery: New York, NY, USA, 2017; pp. 797–806.
12. Tu, K.; Chen, C.; Hou, C.; Yuan, J.; Li, J.; Yuan, X. Rumor2vec: A rumor detection framework with joint text and propagation structure representation learning. *Inf. Sci.* **2021**, *560*, 137–151. [CrossRef]
13. Yuan, Y.; Wang, Y.; Liu, K. Perceiving more truth: A dilated-block-based convolutional network for rumor identification. *Inf. Sci.* **2021**, *569*, 746–765. [CrossRef]
14. Xu, S.; Liu, X.; Ma, K.; Dong, F.; Xiang, S.; Bing, C. Rumor Detection on Microblogs Using Dual-Grained Feature via Graph Neural Networks. In *PRICAI 2021: Trends in Artificial Intelligence*; Pham, D.N., Theeramunkong, T., Governatori, G., Liu, F., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 205–216.
15. Chen, T.; Li, X.; Yin, H.; Zhang, J. Call Attention to Rumors: Deep Attention Based Recurrent Neural Networks for Early Rumor Detection. In *Trends and Applications in Knowledge Discovery and Data Mining*; Ganji, M., Rashidi, L., Fung, B.C.M., Wang, C., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 40–52.
16. Ma, J.; Gao, W.; Wong, K.F. Detect Rumors on Twitter by Promoting Information Campaigns with Generative Adversarial Learning. In Proceedings of the WWW '19: World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 3049–3055. [CrossRef]
17. Yang, F.; Liu, Y.; Yu, X.; Yang, M. Automatic Detection of Rumor on Sina Weibo. In Proceedings of the MDS '12: ACM SIGKDD Workshop on Mining Data Semantics, Beijing, China, 12–16 August 2012; Association for Computing Machinery: New York, NY, USA, 2012. [CrossRef]
18. Shu, K.; Zhou, X.; Wang, S.; Zafarani, R.; Liu, H. The Role of User Profiles for Fake News Detection. In Proceedings of the ASONAM '19: 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, Columbia, VB, Canada, 27–30 August 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 436–439. [CrossRef]
19. Liu, Y.; Wu, Y.F. Early Detection of Fake News on Social Media Through Propagation Path Classification with Recurrent and Convolutional Networks. 2018. Available online: https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/viewPaper/16826 (accessed on 18 August 2021).
20. Castillo, C.; Mendoza, M.; Poblete, B. Information Credibility on Twitter. In Proceedings of the WWW '11: 20th International Conference on World Wide Web, Hyderabad, India, 28 March–1 April 2011; Association for Computing Machinery: New York, NY, USA, 2011; pp. 675–684. [CrossRef]
21. Jin, Z.; Cao, J.; Guo, H.; Zhang, Y.; Luo, J. Multimodal Fusion with Recurrent Neural Networks for Rumor Detection on Microblogs. In Proceedings of the MM '17: 25th ACM International Conference on Multimedia, Mountain View, CA, USA, 23–27 October 2017; Association for Computing Machinery: New York, NY, USA, 2017; pp. 795–816. [CrossRef]
22. Qi, P.; Cao, J.; Yang, T.; Guo, J.; Li, J. Exploiting Multi-domain Visual Information for Fake News Detection. In Proceedings of the 2019 IEEE International Conference on Data Mining (ICDM), Beijing, China, 8–11 November 2019; pp. 518–527. [CrossRef]

23. Shu, K.; Cui, L.; Wang, S.; Lee, D.; Liu, H. DEFEND: Explainable Fake News Detection. In Proceedings of the KDD '19: 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 395–405. [CrossRef]
24. Zhou, K.; Shu, C.; Li, B.; Lau, J.H. Early Rumour Detection. In *Human Language Technologies, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, Minneapolis, MN, USA, 2–7 June 2019*; Volume 1 (Long and Short Papers); Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; pp. 1614–1623. [CrossRef]
25. Bian, T.; Xiao, X.; Xu, T.; Zhao, P.; Huang, W.; Rong, Y.; Huang, J. Rumor Detection on Social Media with Bi-Directional Graph Convolutional Networks. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 549–556. [CrossRef]
26. Ma, J.; Gao, W.; Wong, K.F. Rumor Detection on Twitter with Tree-structured Recursive Neural Networks. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; Volume 1: Long Papers; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 1980–1989. [CrossRef]
27. Lu, Y.J.; Li, C.T. GCAN: Graph-aware Co-Attention Networks for Explainable Fake News Detection on Social Media. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 505–514. [CrossRef]
28. Yahui, L.; Xiaolong, J.; Huawei, S.; Bao, P.; Xueqi, C. A Survey on Rumor Identification over Social Media. *Chin. J. Comput.* **2020**, *41*, 108–130.
29. Shu, K.; Sliva, A.; Wang, S.; Tang, J.; Liu, H. Fake News Detection on Social Media: A Data Mining Perspective. *Sigkdd Explor. Newsl.* **2017**, *19*, 22–36. [CrossRef]
30. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Human Language Technologies, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, Minneapolis, MN, USA, 2–7 June 2019*; Volume 1 (Long and Short Papers); Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; pp. 4171–4186. [CrossRef]
31. Kim, Y. Convolutional Neural Networks for Sentence Classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1746–1751. [CrossRef]
32. Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A.; Hovy, E. Hierarchical Attention Networks for Document Classification. In *Human Language Technologies, Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics, San Diego, CA, USA, 12–17 June 2016;* Association for Computational Linguistics: Stroudsburg, PA, USA, 2016; pp. 1480–1489. [CrossRef]
33. Lu, J.; Yang, J.; Batra, D.; Parikh, D. Hierarchical Question-Image Co-Attention for Visual Question Answering. In *Advances in Neural Information Processing Systems*; Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R., Eds.; Curran Associates, Inc.: New York, NY, USA, 2016; Volume 29.
34. Song, C.; Yang, C.; Chen, H.; Tu, C.; Liu, Z.; Sun, M. CED: Credible Early Detection of Social Media Rumors. *IEEE Trans. Knowl. Data Eng.* **2021**, *33*, 3035–3047. [CrossRef]
35. Yao, L.; Mao, C.; Luo, Y. Graph Convolutional Networks for Text Classification. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 7370–7377. [CrossRef]