information

Article

A Rumor Detection Method Based on Adaptive Fusion of Statistical Features and Textual Features

Ziyan Zhang [1,2], Zhiping Dan [1,2,*], Fangmin Dong [1,2], Zhun Gao [1,2] and Yanke Zhang [1,2]

[1] College of Computer and Information Technology, China Three Gorges University, Yichang 443002, China
[2] Hubei Key Laboratory of Intelligent Vision Based Monitoring for Hydroelectric Engineering, China Three Gorges University, Yichang 443002, China
* Correspondence: zp_dan@ctgu.edu.cn

Abstract: Many rumors spread quickly and widely on social media, affecting social stability. The rumors of most current detection methods only use textual information or introduce external auxiliary information (such as user information and propagation information) to enhance the detection effect, and the inherent statistical features of the corpus have not been fully used and compared with the external auxiliary features; in addition, statistical features are more certain and can only be obtained from textual information. Therefore, we adopted a method based on the adaptive fusion of word frequency distribution features and textual features to detect rumors. Statistical features were extracted by encoding statistical information through a variational autoencoder. We extracted semantic features and sequence features as textual features through a parallel network comprising a convolutional neural network and a bidirectional long-term memory network. In addition, we also designed an adaptive valve to only fuse useful statistical features with textual features according to the credibility of textual features, which can solve the over-fitting problem caused by the excessive use of statistical features. The accuracy of the model in three public datasets (Twitter15, Twitter16, and Weibo) reached 87.5%, 88.6%, and 95.8%, respectively, and the F1 value reached 87.4%, 88.5%, and 95.8%, respectively, showing that the model can effectively improve the performance of rumor detection.

Keywords: rumor detection; adaptive gate; statistical features; textual features

Citation: Zhang, Z.; Dan, Z.; Dong, F.; Gao, Z.; Zhang, Y. A Rumor Detection Method Based on Adaptive Fusion of Statistical Features and Textual Features. *Information* **2022**, *13*, 388. https://doi.org/10.3390/info13080388

Academic Editor: Krzysztof Szczypiorski

Received: 17 July 2022
Accepted: 10 August 2022
Published: 16 August 2022
Corrected: 2 February 2023

1. Introduction

In the Internet era, social media platforms are favored by people for their features of freedom of speech and information sharing. However, because of the advantages of the huge amount of information and fast transmission, a considerable amount of false information and even rumors exist on social media platforms [1]. Allowing rumors to spread willfully will cause immeasurable harm to society [2]. However, because of the limitation of professional knowledge, time, and space, most people cannot accurately detect rumors; thus, a fast and effective method to identify rumors is of great significance to purify the network environment and maintain social stability.

In the early studies of rumor detection, researchers mainly constructed features by hand and used decision trees [3], SVMs [4], random forests [5], and other machine learning methods to classify rumors. These methods rely considerably on feature engineering and require a large number of human resources, and cannot automatically detect rumors. With the rapid development of deep learning, scholars gradually applied some deep learning methods based on RNNs and CNNs for rumor detection [6,7]. Although these methods can realize automatic rumor detection, they only use textual content information and pose the problem of having a single feature, so their detection effect is mediocre. Some subsequent research work introduced external information such as propagation features and user features to assist judgment [8]. Although this development has improved the detection performance, it also has certain shortcomings. First, the inherent and primitive feature of word frequency distribution in the corpus is ignored. Through the comparative analysis of

the rumor corpus, we found that the distribution of some unique words on a certain label is relatively concentrated. For example, when there are multiple emphatic punctuation marks in the input sentence, it is likely to be a rumor, so it is necessary to integrate the statistical feature of word frequency. Second, these works have not found an effective fusion mechanism of external information features and textual content information. In the process of feature fusion, not all textual information needs to be fused with external information. If the textual information itself has rich semantics, introducing a large number of external features will cause over-fitting. Finally, in terms of semantic feature extraction, most of these works use a single-layer semantic feature extraction network, which cannot obtain multidimensional semantic features.

Based on the above problems, we combined the adaptive fusion mechanism of statistical features proposed by Li et al. in 2021 [9] and improved the semantic feature extraction module. In addition, we proposed a new rumor detection method that fuses statistical features with multi-textual features. An adaptive gating module was introduced to control the inflow of statistical features according to the credibility value of textual features, to enhance the rumor classification effect. For the statistical feature extraction module, since the traditional TF-IDF method cannot reflect the positional information and the importance of words, we used a new encoding method to encode the word frequency statistics (Section 3.1), which can provide information on the position of words. Then, we fed it into a variational encoder to obtain a statistical feature representation compatible with semantic features. For the semantic feature extraction module, we input the text into the BERT model and then connected a parallel network of CNN+Attention and Bi_LSTM+Attention to extract semantic feature vectors that consider both local semantic features and global time series features. Additionally, the features were mapped into the information space by the Sigmoid activation function to obtain the credibility of textual features (when the credibility value is close to 0 or 1, the credibility is high; when the reliability value is close to 0.5, the reliability is low); for the adaptive fusion module, a valve component was used to adjust the incoming information flow of statistical features and add auxiliary information to the textual features with low reliability, while the textual features with high reliability remain unchanged, thus achieving a better balance between textual features and statistical features [9]. Our experiments using three public datasets (Twitter15, Twitter16, and Weibo) showed that the proposed model has good rumor detection ability and generalization ability. The main work was performed as follows:

(1) Rumor detection was carried out by fusing word frequency statistical features;

(2) A valve component was introduced to make the model adaptively fuse necessary statistical information;

(3) Eight typical rumor detection models were selected and compared with the model on three datasets to verify the validity of our model for rumor detection;

(4) A large number of ablation experiments were carried out to prove the validity of each module of the model.

2. Related Work

The goal of rumor detection is to judge whether it is a rumor according to the relevant content information posted by users (such as textual content, comments, communication mode, etc.). Current rumor detection methods are mainly divided into (1) designing handcrafted features for rumor detection, (2) extracting textual features for rumor detection, (3) fusing auxiliary features for rumor detection, and (4) rumor detection based on multimodality.

2.1. Design of Handcrafted Features for Rumor Detection

Early studies on rumor detection mainly designed some handcrafted features for rumor detection. In 2011, Castillo et al. combined textual, user-related, topic-related, and communicational features and adopted a decision tree classification model to study the credibility of news on Twitter [3]. In 2012, based on previous studies, Yang et al. combined

features such as the type of publisher's client and the location of event publication and adopted an SVM classifier to conduct rumor detection research on Weibo [4]. In 2013, Known et al. combined temporal, structural, and language features to adopt a random forest classifier for rumor detection [5].

2.2. Extraction of Textual Features for Rumor Detection

With the development of artificial intelligence, some scholars gradually applied deep learning to rumor detection. In 2016, Ma et al. applied deep learning to rumor detection for the first time by extracting textual features using an RNN [6]. In 2017, Yu et al. used a CNN to capture interactions between important features to identify false information [7]. In the same year, Chen et al. integrated an RNN as well as an attention mechanism to extract key features for rumor detection [10]. In 2018, Ajao et al. combined an RNN with an LSTM to extract multidimensional semantic information for real-time rumor detection [11].

2.3. Rumour Detection Based on the Integration of Auxiliary Features

As previously mentioned in Section 2.2, scholars mainly focused on the feature information of the text, but some tweets did not have typical rumor features, so scholars integrated other auxiliary information on textual information for judgment. In 2017, Ruchansky et al. combined user-related, textual, and behavioral features for rumor detection [8]. In the same year, Nguyen et al. used convolutional neural networks combined with sequence information to learn the hidden representation of each tweet [12]. In 2018, Ma et al. obtained more detailed information on events by comparing the structural similarities of different propagation trees [13].

2.4. Rumor Detection Based on Multimodality

With the great success of deep neural networks in learning image and text representations, researchers proposed multimodality-based rumor detection methods. In 2017, Jin et al. proposed a multimodality-based rumor detection model that extracts multimodal information, including visual, textual, and social contextual features, and fuses them through an attention mechanism [14]. In 2019, Khattar et al. proposed a multimodal variational autoencoder to learn shared representations for both texts and images [15]. In 2022, Zhou et al. proposed a multimodality-based multitask and domain-adaptive network that fuses textual and visual representations through two strategies [16].

The traditional machine learning method is characterized by cumbersome engineering and requires a large number of human, material, and financial resources. Although the textual feature extraction method can automatically learn textual features, it cannot recognize some obscure rumors because of only having a single textual feature. It can be derived from previous research that introducing some auxiliary features can improve the performance of rumor detection to a certain extent, but most of them are limited to extracting features other than text and ignore the inherent features on textual datasets (word frequency distribution features on label datasets). In addition, the extraction of textual features is also of great significance for multimodal rumor detection. Most multimodal information needs to integrate textual content information. Therefore, undoubtedly, improvement in the detection effect of textual information will lead to a significantly positive effect on multimodal methods.

Thus, we adopted a method based on the adaptive fusion of statistical features to detect rumors and improve the performance of rumor detection.

3. Materials and Methods

Our model mainly consists of four parts: statistical feature extraction, textual feature extraction, feature adaptive fusion, and classification. We show the overall model structure in Figure 1.

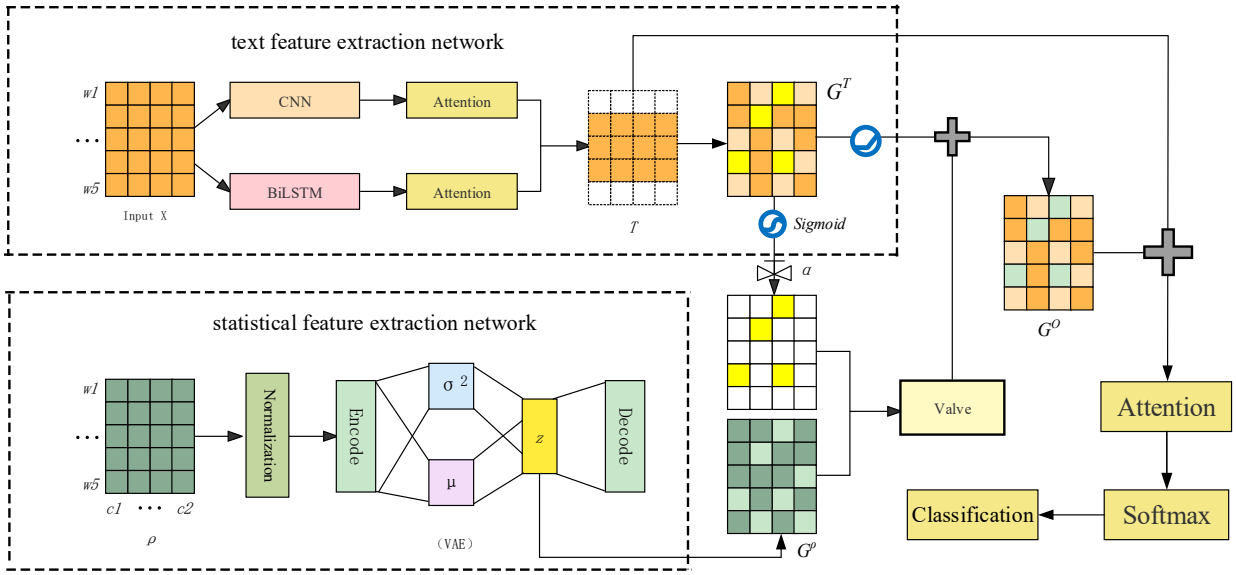

Figure 1. Overall model structure.

3.1. Word Frequency Statistical Vector Representation

We define ***Fc*** as the statistical vector of word frequency.

If w is used to represent a word, and c is used to represent the type of tag, the word frequency statistic vector ***Fc*** of w can be expressed as

$$\rho^{\mathrm{w}} = [\rho_1, \ldots, \rho_c] \tag{1}$$

where ρ_i is the distribution of the word w on label i.

If S is used to represent a tweet, and m is used to represent the length of the tweet, then the word frequency statistic vector ***Fc*** of tweet S is expressed as

$$\rho^{\mathrm{s}} = [\rho^{w_1}, \ldots, \rho^{w_m}] \tag{2}$$

where ρ^{w_i} is the word frequency statistic vector of the i-th word.

Since the word frequency statistics vectors of tweets are combined in order from the word frequency statistics vectors of the words, it can effectively reflect the positional distribution of each word. It should be noted that the word frequency statistics were obtained only from the training set. The word frequency distribution feature on datasets is original and inherent, but some specific words have limited contributions to the classification task. For example, if the word w appears in all labels with a high or low frequency, we consider its role being limited. In contrast, if the word w appears frequently only on one type of label, it is considered more important.

3.2. Statistical Feature Extraction Network

Since the original ***Fc*** is dimensionally incompatible with the semantic feature vector, a variational autoencoder is used to map the discrete ***Fc*** into a continuous space and convert the statistical features into an efficient vector representation [9]. We show its model structure in Figure 2.

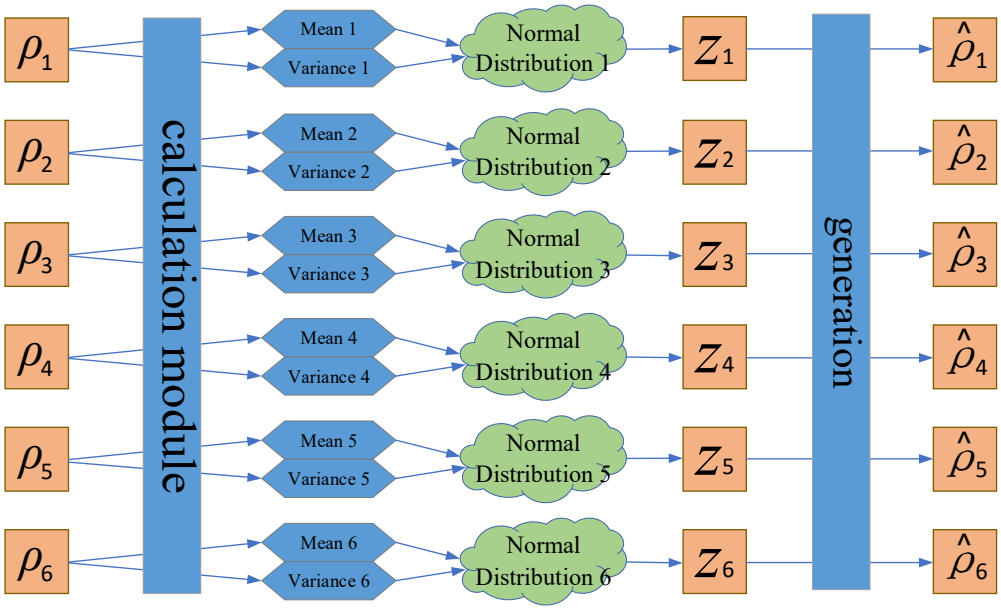

Figure 2. Structure of variational autoencoder.

Assuming that the random process p_θ $(\boldsymbol{\rho} \mid \boldsymbol{z})$ generates ***Fc*** from all the data in the datasets, we obtain

$$z = \left\{ \rho_{(i)}^{s} \right\}_{i=1}^{N} \tag{3}$$

where N is the size of the datasets, and the process involves the latent variable $\boldsymbol{z}$ sampled from the prior distribution p $(\boldsymbol{z})$. However, the posterior distribution p_θ $(\boldsymbol{z} \mid \boldsymbol{\rho})$ itself is not good to ask, so the construction of deep learning network learning parameters θ and φ, through step-by-step optimization variational approximation function p_φ $(\boldsymbol{z} \mid \boldsymbol{\rho})$, result in the infinite p_θ $(\boldsymbol{\rho} \mid \boldsymbol{z})$.

$$L = log(p_\theta(\rho)) = L^V + D_{KL}(q_\varphi(z|\rho)||p_\theta(z|\rho)) \tag{4}$$

where D_{KL} is KL divergence, which is used to measure the similarity between p and q distributions. The smaller the value is, the closer the two distributions are; otherwise, the larger the gap becomes. Since D_{KL} is non-negative, L^V is the lower bound of variation of L.

$$L^V = -D_{KL}(q_\varphi(z|\rho)||p_\theta(z|\rho)) + E_{q_\varphi(z|\rho^{(i)})}(log(p_\theta(\rho^{(i)}|z))) \tag{5}$$

In the sampling process, the technique of reparameterization is used, which makes the posterior of the latent variable $\boldsymbol{z}$ obey a standard normal distribution p_θ $(\boldsymbol{\rho} \mid \boldsymbol{z})$ for each sample. Two encoders are used in this model, one for calculating the mean μ and the other for calculating the variance σ^2. Since the approximate prior is multivariate Gaussian, we represent the variational posterior using a diagonal covariance structure.

$$log_{q_\varphi}(z|\rho) = logN(z; \mu, \sigma^2 I) \tag{6}$$

By using VAE for unsupervised training, the coded latent variable $\boldsymbol{\rho^z}$ is obtained, which incorporates the global information of the statistical vector. This module is independent of the textual feature extraction module, and it adaptively fuses with textual features through a valve component and then inputs them into the classifier.

3.3. Textual Feature Extraction Network

The network's function is to extract textual features from incoming tweets. This network is mainly divided into three parts: the BERT layer, the Bi_LSTM+Attention layer, and the CNN+Attention layer. We show the model diagram in Figure 3.

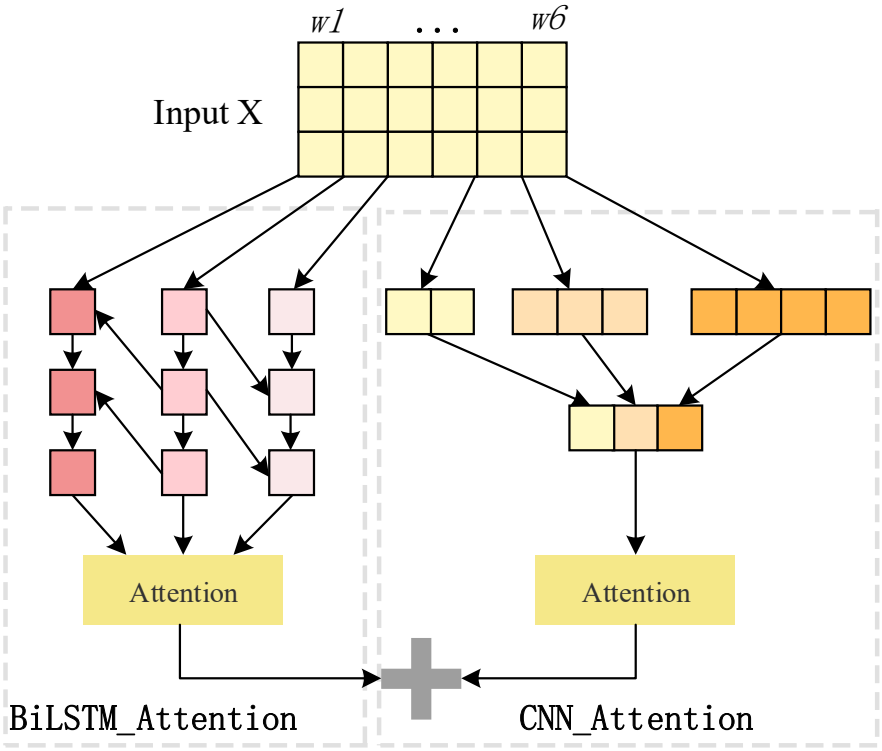

Figure 3. Network structure diagram of textual feature extraction.

3.3.1. BERT

The tweet text of length m is sent to the BERT model for pretraining, and the sentence vector $\boldsymbol{X}$ is output after encoding. If $x_j \in R^d$ is the d-dimension embedding of the j-th word, then

$$X_{1:m} = [x_1; x_2; \ldots; x_m] \tag{7}$$

3.3.2. Bi_LSTM+Attention Layer

LSTM is a variant of RNN, which is composed of an input gate, a forgetting gate, an output gate, and an internal memory unit. It can capture long-distance global features and temporal features and effectively ease gradient disappearance and gradient explosion. However, it can only encode backward information, not forward information. For a tweet, its contextual influence is important. Therefore, we use Bi_LSTM to capture the sequence characteristics of its context:

$$H = [\overrightarrow{h}, \overleftarrow{h}] = BiLSTM(x) \tag{8}$$

For each tweet, the key degree of each word is different. Introducing an attention mechanism can assign different weights to each word: The more important the word is, the higher its weight becomes, and the less important the word is, the lower its weight is; this can further improve the model's ability to extract key features. The calculation process is expressed in Equations (9)–(11):

$$u = \text{Tanh}(W^H H + b^H) \tag{9}$$

$$a_i = Softmax(u^T, u_s) \tag{10}$$

$$T_1 = \sum_t a_i H \tag{11}$$

where $\boldsymbol{W}$ is the weight matrix, $\boldsymbol{b}$ is the bias term, a_i is the distribution weight of attention, and $\boldsymbol{T_1}$ is the output of the Bi_LSTM layer.

3.3.3. CNN+Attention Layer

A CNN is composed of a convolution layer and a pooling layer. It is good at extracting local features of textual information. In the method used in this paper, a convolution kernel of size (3, 4, 5) is used to perform convolution and maximum pooling operations on the input sentence vector, and then the obtained feature vectors are horizontally spliced to generate a new feature vector $\boldsymbol{K}$ that fuses different semantic information.

$$K = CNN(x) \tag{12}$$

Then, $\boldsymbol{K}$ is taken as the input of the attention layer and the output $\boldsymbol{T_2}$ of the CNN+Attention layer is obtained using Equations (9)–(11). Not only does it contain rich semantics, but it also highlights key information.

Finally, two weak learning models with large differences and complementary functions are combined through a parallel strategy to obtain a strong learning model, which obtains textual features that take into account both the local semantics and global sequence features of the text. Specifically, the output $\boldsymbol{T_1}$ of the Bi_LSTM+Attention layer and the output $\boldsymbol{T_2}$ of the CNN+Attention layer are spliced to obtain the textual feature $\boldsymbol{T}$ fused with multiple features; then, a dense layer is used to map $\boldsymbol{T}$ into an information space to obtain $\boldsymbol{G^T}$, the Sigmoid function is used to activate $\boldsymbol{G^T}$, and G'^T is then used to evaluate the credibility of textual features.

$$G'^T = Sigmoid(G^T) = Sigmoid(W^T T + b^T) \tag{13}$$

3.4. Valve Component

To flexibly apply statistical features, a dense layer is used to project $\boldsymbol{\rho^z}$ into a space shared with textual information to obtain $\boldsymbol{G^\rho}$.

$$G^\rho = W^\rho(\rho^z) + b^\rho \tag{14}$$

The AdaGate function is used to fuse $\boldsymbol{G^T}$ and $\boldsymbol{G^\rho}$, and the enhanced textual feature $\boldsymbol{G^o}$ is output fused with the statistical feature $\boldsymbol{G^\rho}$ [9].

$$G^o = AdaGate(G^T, G^\rho, G'^T, \alpha) = ReLU(G^T) + Valve(G'^T, \alpha) \otimes G^\rho \tag{15}$$

where ReLU is the activation function, $\otimes$ is the adaptive fusion valve, and G'^T has a probability value between 0 and 1; when it is close to 0 or 1, the credibility is high reliability, and when it is close to 0.5, the credibility is low. The role of the valve function is to match useful statistics $\boldsymbol{G^\rho}$ to low-confidence text (i.e., G'^T close to 0.5). α is a hyperparameter used to adjust the threshold of confidence. When α = 0, all statistical features are discarded; when α = 0.5, all statistical features are fused. Therefore, the adaptive fusion valve utilizes valve (G'^T, α) as a filter to provide only the necessary information.

3.5. Classifier

We use an attention mechanism to combine the augmented textual features $\boldsymbol{G^\circ}$ and the original features $\boldsymbol{T}$.

$$Attention(G^o, T) = Softmax(G^o T^T)T \tag{16}$$

When statistical features are completely discarded (i.e., α = 0), $\boldsymbol{G^\circ} = \boldsymbol{T}$, then formula (16) turns into a self-attention mechanism [16].

Finally, after the fully connected layer and the SoftMax layer are determined, the final vector representation is projected into the label space, and the cross-entropy loss function is used as the optimizer for rumor classification.

$$L = -\sum_{C \in N} \left(y_c \log(y_c) + (1 - y_c) \log\left(1 - \hat{y_c}\right) \right) \quad (17)$$

where L is the loss function, N is the set of all tweets, y_c is the true label of tweet c, and $\hat{y_c}$ is the predicted label of tweet c.

4. Results

4.1. The Datasets

We used three public datasets, namely Twitter15 [17], Twitter16 [17], and Weibo [6], for the experiments. Their data were obtained from the most popular social networking sites in China and the United States, respectively.

We used a two-category label in the Weibo dataset, namely "rumor" and "non-rumor"; and four-category labels in Twitter15 and Twitter16, namely "false rumor", "true rumor", "non-rumor", and "unverified", among which both "false rumors" and "true rumors" are rumors, the difference is that "false rumors" represent some negative rumors, and "true rumors" represent some positive rumors. The information statistics of the three public datasets are listed in Table 1.

Table 1. Datasets information statistical table.

	Weibo	**Twitter15**	**Twitter16**
False rumors	2313	374	205
Non-rumors	2351	370	205
unverified	0	374	203
True rumor	0	372	205
Total	4664	1490	818

In order to reduce the influence of the absence of words in the data on the experiment, we used a regular expression to clean the dataset and divided the dataset into a training set and a verification set in a ratio of 4:1.

4.2. Training Parameter Setting

The Chinese and English BERT model [18] used in this paper contains 12 layers of transformer and its dimension of word embedding is 768; the size of the convolution kernel in the CNN was set to (3, 4, 5), each with 100 volumes of accumulation kernels; in the Bi_LSTM model, our hidden layer size was set to 128; all deep learning frameworks used in this paper were TensorFlow; all experiments were performed on NVIDIA 2080 ti.

The variational auto-encoder network's training parameters are shown in Table 2.

Table 2. Variational auto-encoder network's training parameters.

The Parameter Name	**The Parameter Value**
Batch size	32
Training epoch	20
Hidden layer size	128
Optimizer	Adam
Loss function	VAE_LOSS

Multiclassifier training parameters are shown in Table 3.

Table 3. Multiclassifier training parameters.

The Parameter Name	The Parameter Value
Batch size	32
Training epoch	10
Optimizer	Adam
Loss function	Cross entropy loss
Learning rate	0.05
Dropout	0.2

4.3. Comparative Experiment and Result Analysis

In order to prove the effectiveness of the adaptive gating network fused with statistical features for rumor detection, we compared our model with the following 10 models:

(1) DTR [19]: a decision tree model for detecting fake news by query phrases;

(2) DTC [3]: a decision tree model using news feature combination;

(3) RFC [5]: a random forest classifier using user-related, language, and structural features;

(4) PTK [17]: an SVM classifier with propagation tree kernels to detect fake news by learning the temporal structure from propagation;

(5) GRU [6]: an RNN-based model that learns language patterns from user reviews;

(6) RvNN [13]: a recurrent neural network based on tree structures, with a GRU unit that learns rumor representation through propagation structures;

(7) PPC [20]: a new model to detect fake news by combining the propagation path classification of recurrent networks and convolutional networks;

(8) HD—TRANS [21]: a model based on tree transformer networks, which focuses on proving its validity in shallow and deep conversations of datasets, respectively;

(9) BDCoNN [22]: a rumor detection method that combines user-related, content-related, and commenting features.

(10) BERT_fine-tuning: a method that adopts the directly fine-tuned BERT model and the word frequency statistics proposed in our paper for its adaptive fusion.

Since the label distribution of the datasets may be unbalanced, it is not enough to use the accuracy rate as the evaluation index to measure the performance of the model, so in this paper, we used the accuracy rate and the macro average F1 value as the evaluation index. The experimental results are presented in Table 4, with the best results for the columns emphasized in bold.

Table 4. Comparative experimental results.

	Twitter15		**Twitter16**		**Weibo**	
Model	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**
DTR	0.467	0.443	0.566	0.515	0.732	0.732
DTC	0.523	0.502	0.538	0.497	0.831	0.831
RFC	0.599	0.55	0.582	0.533	0.849	0.847
PTK	0.75	0.75	0.732	0.743	/	/
RvNN	0.749	0.742	0.737	0.704	/	/
HD-TRANS	0.789	0.787	0.768	0.765	/	/
GRU	0.646	0.642	0.633	0.635	0.91	0.91
PPC	0.842	0.824	0.863	0.850	0.921	0.921
BERT_fine-tuning	0.847	0.847	0.856	0.855	0.951	0.949
BDCoNN	/	/	/	/	0.957	0.957
BCBA_GN	**0.875**	**0.874**	**0.886**	**0.885**	**0.958**	**0.958**

From the data in the table, it can be seen that the DTR, DTC, and RFC models based on handcrafted features performed significantly worse due to their inability to accurately capture useful features; RFC performed relatively well due to its use of additional sequence features. The three tree-based models, namely PTK, RvNN, and HD-TRANS, outperformed the handcrafted, feature-based models by a large margin because they use propagation trees to extract features related to language structure. In the deep learning-based models, GRU, PPC, BERT_fine-tuning, and BDCoNN, the neural network automatically learns latent features, but because PPC relies on the fixed user features of the forwarding sequence and combines CNNs and RNNs to capture changes in user-related feature, its performance was better than GRU and BERT_fine-tuning. BDCoNN achieved the best performance on Weibo due to integrating user-related, content-related, and commenting features and its full use of an attention mechanism.

The proposed model outperformed all baseline models in performance. Regarding the Twitter15 and Twitter16 datasets [15], compared with the best-performing PPC, the accuracy increased by 3.3% and 2.3%, respectively; the F1 value increased by 5% and 3.5%, respectively. On the Weibo dataset, compared with the best-performing BDCoNN, the accuracy and F1 value improved by 0.1%, respectively. These results confirmed the effectiveness of fusing statistical information with adaptive valves.

4.4. Variational Autoencoders (VAE) Compared with Ordinary Auto-Encoders (AE)

Both VAE and AE are powerful generative networks [9]. We used variational auto-encoders and ordinary auto-encoders to encode the word frequency statistics vector ***Fc*** and send it into the model, taking $\alpha = 0.25$ for experiments. The results are presented in Table 5. Since the AE maps the input to a numerical code, it over-fit during the training process, and the generalization was poor. By contrast, the VAE maps the input to a distribution set and introduces the *KL* divergence to improve generalization ability. Therefore, the performance of the VAE network was better, and the accuracy rates on Twitter15, Twitter16, and Weibo datasets increased by 0.2%, 0.4%, and 0.9%, respectively; the F1 value increased by 0.3%, 0.3%, 0.9%. These results revealed that using variational auto-encoders is more conducive to the representation learning of statistical features and can improve the classification performance of the model.

Table 5. Variational auto-encoders compared with ordinary auto-encoders.

	VAE		**AE**	
	Acc	**F1**	**Acc**	**F1**
Twitter15	0.851	0.849	0.849	0.846
Twitter16	0.886	0.885	0.882	0.879
Weibo	0.952	0.952	0.943	0.943

5. Ablation Study

5.1. Valve Components

In the adaptive valve component, the improvement effect of word frequency statistical features on the model is affected by the hyperparameter α, which determines the credibility threshold for triggering information fusion. To study the influence of different α on the model, we conducted experiments on three datasets Twitter15, Twitter16, and Weibo, respectively. From the results in Figure 4, it can be seen that the adaptive fusion of some statistical features was better than completely discarding statistical features in terms of accuracy and F1 value ($\alpha = 0$) and fusing all statistical features ($\alpha = 0.5$). Therefore, not all statistical features are useful, and adding some unnecessary statistical features causes noise to the classification and affects the classification effect. At the same time, the experiment also proved that adding valve components can effectively solve the over-fitting problem caused by the excessive use of statistical features.

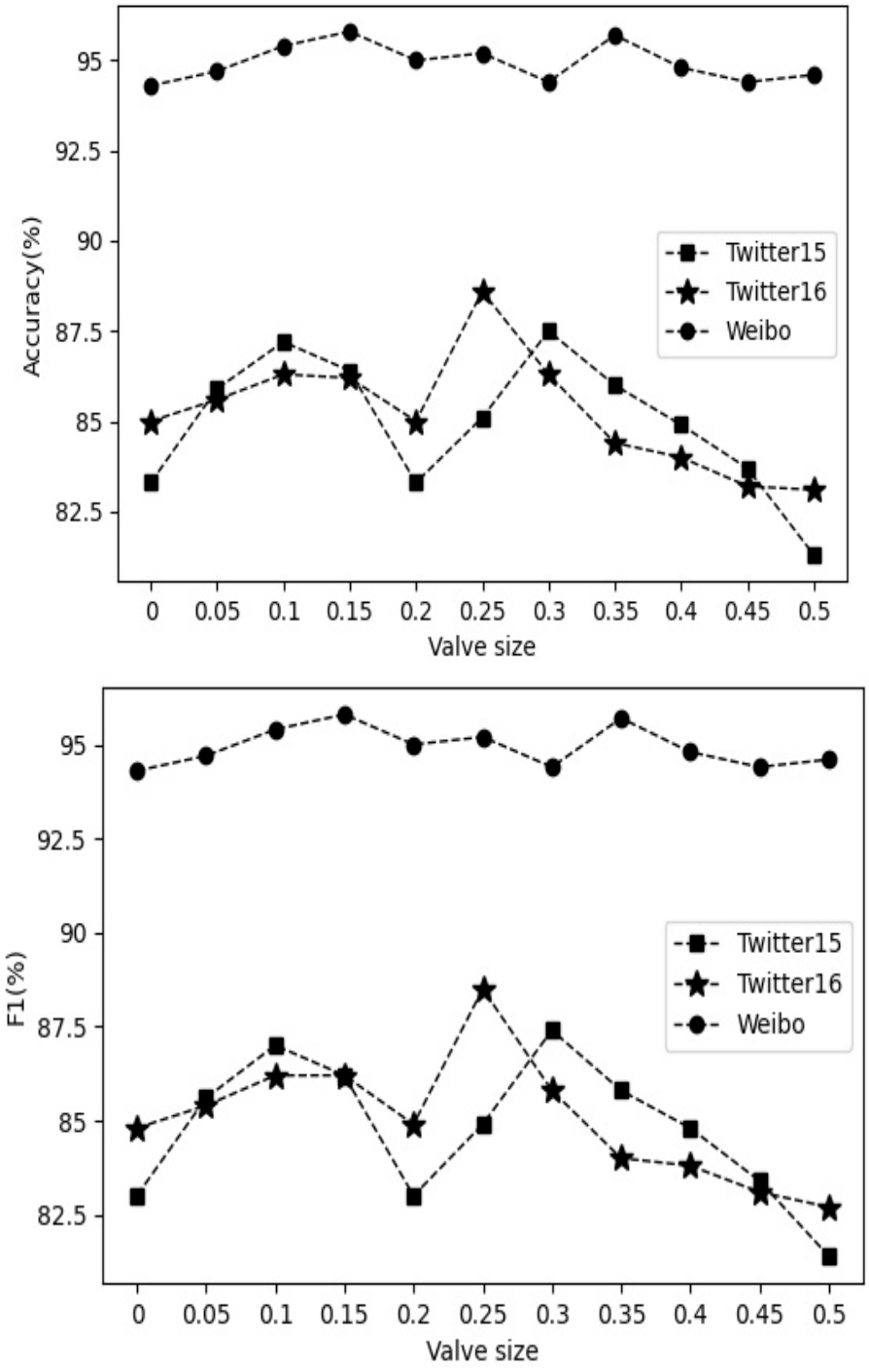

Figure 4. Influence of different valve thresholds on model performance.

5.2. Textual Feature Module

Since the influence of statistical features on different datasets was different, we chose to completely discard statistical features ($\alpha = 0$) to conduct ablation research on the textual feature extraction part. We show the experimental results in Figure 5.

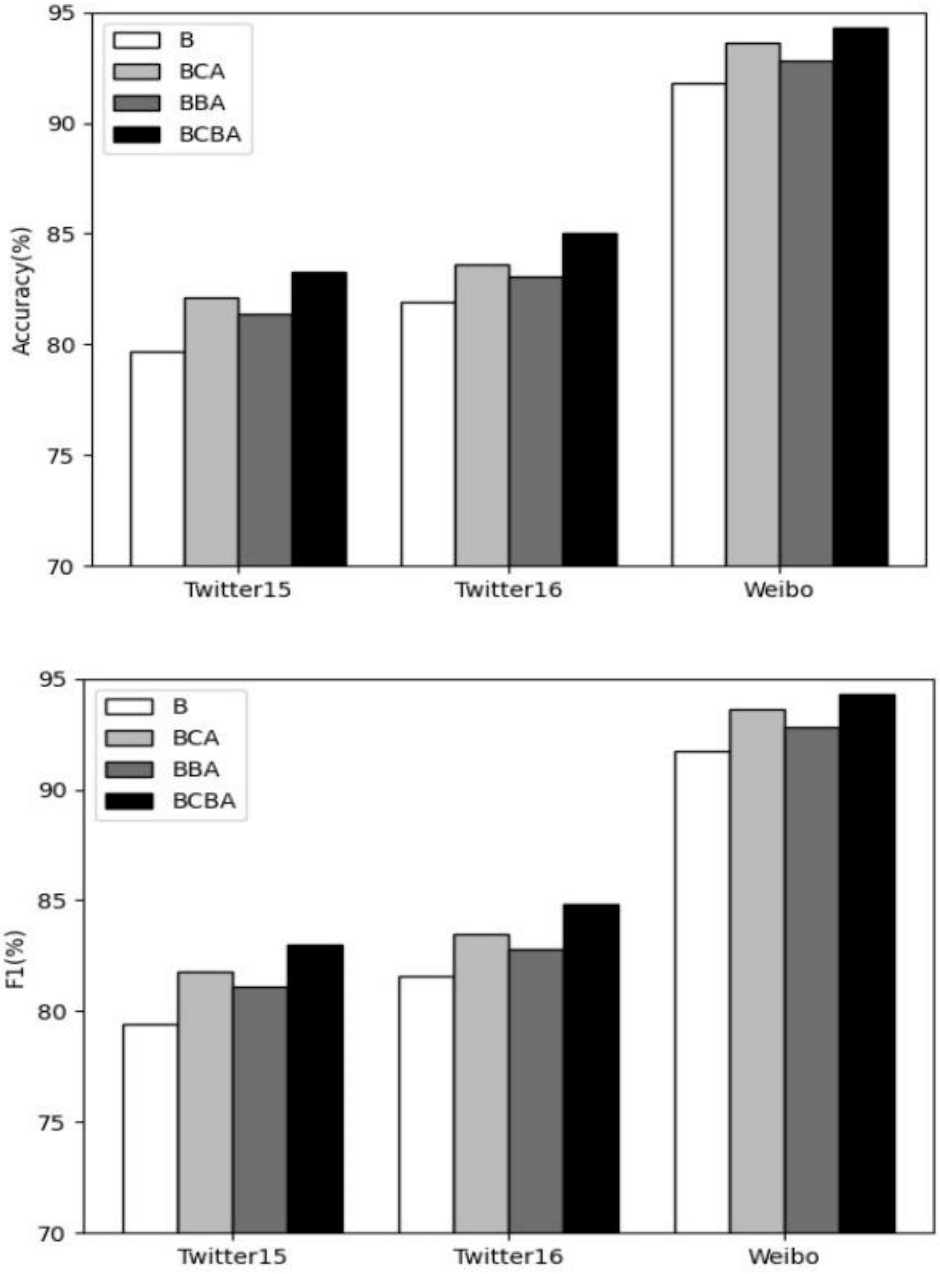

Figure 5. Results of semantic feature extraction ablation.

B: Only the BERT model was used for textual feature extraction.

BBA: the CNN+Attention module was removed, and only the BERT and Bi_LSTM+Attention models were used for textual feature extraction.

BCA: the Bi_LSTM+Attention module was removed, and only the BERT and CNN+Attention models were used for textual feature extraction.

BCBA: the BERT model was used for pretraining and sent to the CNN+Attention and Bi_LSTM+Attention parallel networks for textual feature extraction, which is the method we used.

It can be seen from the results in the above figure that, on the basis of the BERT model, the effect of adding the CNN+Attention module or the Bi_LSTM+Attention module was improved to a certain extent; compared with the sequence features of the text, its semantic features could more directly mine the text. Therefore, the effect of adding the CNN+Attention module alone on the three datasets was better than adding the Bi_LSTM+Attention module

alone; we fused the CNN+Attention and Bi_LSTM+Attention modules to take into account the textual semantic features and sequence features to achieve the best results.

6. Conclusions

We proposed a rumor detection method based on the adaptive fusion of statistical features and textual features. A VAE was used to encode statistical features combined with an attention mechanism, which can not only capture key information but also highlight locational information. The CNN+Attention and Bi_LSTM+Attention parallel networks were used to extract textual features, and both semantic features and sequence features were considered. Finally, an adaptive valve component was used to fuse useful statistical information with textual information to avoid the over-fitting problem caused by the excessive use of statistical feature bands. This method had remarkable results on three public datasets.

In future work, we plan to introduce a rumor transmission structure into the model to further enhance the learning ability of the model.

Author Contributions: Conceptualization, Z.D. and Z.Z.; methodology, Z.Z.; validation, Z.Z.; data curation, Z.G.; visualization, Y.Z.; supervision, F.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the China Government under NSFC-Xinjiang Joint Fund (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 first author.

Acknowledgments: This work was supported in part by the National Natural Science Foundation of China (Grant No. U1703261). The corresponding author is Zhiping Dan.

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

References

1. Gao, Y.J.; Liang, G.; Jiang, F.T.; Xu, C.; Yang, J.; Chen, J.R.; Wang, H. Social Network Rumor Detection: A Survey. *Acta Electonica Sin.* **2020**, *48*, 1421.
2. Ma, J.; Gao, W.; Wong, K.F. Detect rumors on twitter by promoting information campaigns with generative adversarial learning. In *WWW '19: The World Wide Web Conference*; ACM: New York, NY, USA, 2019; pp. 3049–3055.
3. Castillo, C.; Mendoza, M.; Poblete, B. Information credibility on twitter. In Proceedings of the 20th International Conference on World Wide Web, Hyderabad, India, 28 March–1 April 2011; pp. 675–684.
4. Yang, F.; Liu, Y.; Yu, X.; Yang, M. Automatic detection of rumor on sina weibo. In Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics, Beijing, China, 12–16 August 2012; pp. 1–7.
5. Kwon, S.; Cha, M.; Jung, K.; Chen, W.; Wang, Y. Prominent features of rumor propagation in online social media. In Proceedings of the 2013 IEEE 13th International Conference on Data Mining, Dallas, TX, USA, 7–10 December 2013; pp. 1103–1108.
6. 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 25th International Joint Conference on Artificial Intelligence (IJCAI 2016), New York, NY, USA, 9–15 July 2016.
7. Yu, F.; Liu, Q.; Wu, S.; Wang, L.; Tan, T. A Convolutional Approach for Misinformation Identification. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17), Melbourne, Australia, 19–25 August 2017; pp. 3901–3907.
8. 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; pp. 797–806.
9. Li, X.; Li, Z.; Xie, H.; Li, Q. Merging statistical feature via adaptive gate for improved text classification. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual Event, 2–9 February 2021; pp.13288–13296. Available online: https://aaai.org/Conferences/AAAI-21/ (accessed on 11 January 2023).
10. 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—PAKDD 2018*; Springer: Cham, Switzerland, 2018; pp. 40–52.

11. Ajao, O.; Bhowmik, D.; Zargari, S. Fake news identification on twitter with hybrid cnn and rnn models. In Proceedings of the 9th International Conference on Social Media and Society, Copenhagen, Denmark, 18–20 July 2018; pp. 226–230.
12. Nguyen, T.N.; Li, C.; Niederee, C. On early-stage debunking Rumors On Twitter: Leveraging the wisdom of weak learners. In *Social Informatics—SocInfo 2017*; Springer: Cham, Switzerland, 2017; pp. 141–158.
13. 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 (ACL 2018), Melbourne, Australia, 15–20 July 2018.
14. 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 25th ACM International Conference on Multimedia, Mountain View, CA, USA, 23–27 October 2017; pp. 795–816.
15. Khattar, D.; Goud, J.S.; Gupta, M.; Varma, V. Mvae: Multimodal variational autoencoder for fake news detection. In *WWW '19: The World Wide Web Conference*; ACM: New York, NY, USA, 2019; pp. 2915–2921.
16. Zhou, H.; Ma, T.; Rong, H.; Qian, Y.; Tian, Y.; Al-Nabhan, N. MDMN: Multi-task and Domain Adaptation based Multi-modal Network for early rumor detection. *Expert Syst. Appl.* **2022**, *195*, 116517. [CrossRef]
17. Ma, J.; Gao, W.; Wong, K.F. Detect rumors in microblog posts using propagation structure via kernel learning. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, BC, Canada, 30 July–4 August 2017.
18. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
19. Zhao, Z.; Resnick, P.; Mei, Q. Enquiring minds: Early detection of rumors in social media from enquiry posts. In Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, 18–22 May 2015; pp. 1395–1405.
20. Liu, Y.; Wu, Y.F. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
21. Ma, J.; Gao, W. Debunking Rumors on Twitter with Tree Transformer. In Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain, 8–13 December 2020.
22. 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. [CrossRef]