

Article

# A Novel Hybrid Model for Cantonese Rumor Detection on Twitter

Xinyu Chen <sup>†</sup> , Liang Ke <sup>†</sup>, Zhipeng Lu <sup>†</sup>, Hanjian Su <sup>†</sup> and Haizhou Wang <sup>\*</sup>

College of Cybersecurity, Sichuan University, Chengdu 610064, China; 2017141531040@stu.scu.edu.cn (X.C.); 2017141531066@stu.scu.edu.cn (L.K.); 2017141491001@stu.scu.edu.cn (Z.L.); 2017141531010@stu.scu.edu.cn (H.S.)

<sup>\*</sup> Correspondence: whzh.nc@scu.edu.cn

<sup>†</sup> These authors contributed equally to this work.

Received: 31 July 2020; Accepted: 26 August 2020; Published: 12 October 2020



**Abstract:** The development of information technology and mobile Internet has spawned the prosperity of online social networks. As the world's largest microblogging platform, Twitter is popular among people all over the world. However, as the number of users on Twitter increases, rumors have become a serious problem. Therefore, rumor detection is necessary since it can prevent unverified information from causing public panic and disrupting social order. Cantonese is a widely used language in China. However, to the best of our knowledge, little research has been done on Cantonese rumor detection. In this paper, we propose a novel hybrid model XGA (namely XLNet-based Bidirectional Gated Recurrent Unit (BiGRU) network with Attention mechanism) for Cantonese rumor detection on Twitter. Specifically, we take advantage of both semantic and sentiment features for detection. First of all, XLNet is employed to produce text-based and sentiment-based embeddings at the character level. Then we perform joint learning of character and word embeddings to obtain the words' external contexts and internal structures. In addition, we leverage BiGRU and the attention mechanism to obtain important semantic features and use the Cantonese rumor dataset we constructed to train our proposed model. The experimental results show that the XGA model outperforms the other popular models in Cantonese rumor detection. The research in this paper provides methods and ideas for future work in Cantonese rumor detection on other social networking platforms.

**Keywords:** online social networks; rumor detection; Cantonese; XGA model

## 1. Introduction

With the rapid development of the Internet, social media have provided a convenient online platform for users to obtain information, express opinions, and communicate with each other. As one of the most popular social networks and microblogging platforms, Twitter has attracted more and more people to publish and share their opinions. As more and more people participate in discussions about hot topics and exchange their opinions on social networks, many rumors appear on Twitter. A rumor refers to a story or statement in general circulation without confirmation or certainty as to facts. Due to the large number of users and easy access to social networks, rumors can spread widely and quickly on Twitter, causing public panic and disrupting social order, even endangering national security [1]. Therefore, it is necessary to detect rumors and stop them from spreading widely on Twitter. There are some fact-checking websites, such as Snopes [2] and PolitiFact [3], that provide information and reports for rumor analysis and checking. However, these websites rely heavily on manual labor to track and debunk rumors, and the verified rumors are often limited to specific topics.

In this case, automatic rumor detection methods should be proposed for saving human effort and debunking rumors more efficiently.

There has been a lot of work related to rumor detection on Twitter concerning English and Chinese [4–7], but Cantonese rumors are seldomly studied. Cantonese is a branch of Chinese and it is mostly used in Guangdong Province and Hong Kong in China. Although Cantonese is widely used in speaking, it is in increasing use for informal communication like in forums or blogs. However, due to the complexity of Cantonese semantics and the lack of benchmark datasets, Cantonese rumor detection is a challenging task.

To address the problem of Cantonese rumor detection, we have proposed a novel hybrid model XGA and conducted experiments to evaluate its performance. The main contributions of this paper are summarized as follows:

- To cope with the complexity of Cantonese semantics and extract the deep features of Cantonese, we take both semantic and sentiment features into account for rumor detection. An XLNet model is used to produce text-based and sentiment-based embeddings. We perform joint learning of character and word embeddings to better fit in with the structure of Cantonese.
- A novel hybrid model XGA is proposed to improve the performance of Cantonese rumor detection, which takes advantage of XLNet, BiGRU and the attention mechanism. The Cantonese rumor dataset we constructed before is used to train our proposed model. The evaluation results show that XGA significantly outperforms other widely used rumor detection approaches in the Cantonese rumor detection.

The rest of this paper is organized as follows. Section 2 introduces related work on rumor detection. Section 3 elaborates the proposed model. Section 4 describes the experimental setup and gives out evaluation results. Section 5 concludes the research in this paper and discusses future work.

## 2. Related Work

Currently, most rumor detection methods are supervised. The literature related to rumor detection has been reviewed by several comprehensive surveys [8–10]. In our work, we briefly review some significant works based on deep learning methods.

Deep learning has been the most revolutionary development in artificial intelligence, which is widely used in the field of rumor detection. The two most popular deep learning models are Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN).

For identifying rumors, Ma et al. [4] utilized RNN to learn continuous representations of microblog events, which captured the variation of contextual information of relevant posts over time. Then, in [11], the authors improved this approach by combining the attention mechanism with RNN to focus on textual features with different attentions. Recently, Ma et al. [12] employed the adversarial learning method to improve the performance of the rumor classifier, where the discriminator was used as a classifier and the corresponding generator improved the discriminator by generating conflicting noises. In addition, a multi-task learning approach was proposed by Kochkina et al. [13] to solve the problem of rumor classification. To be specific, they implemented a multi-task learning framework with an LSTM layer shared among all tasks, as well as a number of task-specific layers. In [14], Sumeet et al. designed a Tree LSTM model for rumor and stance detection that converted the propagation tree into a binarized constituency tree structure. The model applied convolution units in Tree LSTMs, which were better at learning patterns in features and employed the multi-task learning to propagate the useful stance signal up in the tree at the root node. But RNN was not qualified for the early detection of misinformation and had a bias towards the latest input elements.

So in [15], Yu et al. proposed a method based on CNN to learn key features scattered among an input sequence and shape high-level interactions among significant features. What's more, Qian et al. [16] introduced a Two-Level CNN with User Response Generator (TCNN-URG) where TCNN captured underlying semantic information at word and sentence levels, and URG generated

user responses to new articles with the assistance of historical user responses. Furthermore, a mixture of RNNs and CNNs was exploited in recent works. Liu et al. [6] incorporated both RNN and CNN to get the user features based on time series. In addition, a model of Credible Early Detection (CED) was presented by Song et al. [17] to detect rumors on social media based on repost information. CNN was leveraged to obtain feature vectors of original microblogs and repost sequence. Then, the repost sequence was sent into RNN. What's more, in [18], the authors proposed a multi-modal network comprising CNN and Long Short-Term Memory (LSTM) with the attention mechanism. It jointly learned representations of textual contents and social contexts in rumors. However, these methods were inefficient to learn the features of the propagation structure, and they ignored the global structural features of rumor dispersion.

So, to focus on the differences between the characteristics in propagation of real and false information, Ma et al. [7] proposed models based on top-down and bottom-up tree-structured Recursive Neural Networks (RvNN), which deeply integrated the structural and textual features of tweets for detecting rumors at early stages from propagation trees or networks. Based on this, they designed discriminative attention mechanisms for the RvNN-based models to selectively attend on the subset of evidential posts during the bottom-up/top-down recursive composition [19]. Moreover, in [20], the authors built a model based on Bi-Directional Graph Convolutional Network (Bi-GCN) to explore characteristics by operating on both top-down and bottom-up propagation of rumors. But these methods only detected rumors based on the meaning of text and ignored the sentiment of it. In [21], authors designed a hybrid framework to analyze the data from social media based on sentiment analysis. Inspired by this, we proposed a hybrid model that took both semantic and sentiment features into account for Cantonese rumor detection.

### 3. The Proposed Model

In this work, we propose a novel deep neural network-based model XGA to detect Cantonese rumors on Twitter. The XGA takes advantage of semantic and sentiment features for detection. To be specific, XLNet is used to produce text-based and sentiment-based embeddings. In addition, BiGRU and the attention mechanism are involved to extract important semantic features. The structure of the XGA model is shown in Figure 1.

#### 3.1. Input Layer

In this work, we use the Cantonese Rumor Dataset (CR-Dataset) [22] we constructed before. It contains 13,000 tweets, including 6377 rumors and 6623 non-rumors. As shown in *Input Layer* in Figure 1, the input of the model  $I = \{I_1, I_2, \dots, I_n\}$  is a pre-processed tweet. Since the maximum length of a tweet written in Cantonese is 140, the maximum sequence length is set to 140 to cover the input. Then we use tokens to represent each of the characters in the tweet and feed them into XLNet-Text and XLNet-Sentiment, respectively.

#### 3.2. Embedding Layer

As shown in *Embedding Layer* in Figure 1, the model generates text-based and sentiment-based embeddings to extract the semantic and sentiment features of the tweet. In addition, we combine the character embeddings produced by XLNet with Cantonese word embeddings to learn the contextual information and internal structures of the words, so as to make the model more suitable for the Cantonese rumor detection.

XLNet is a generalized Autoregressive (AR) pre-training method that combines the advantages of AR and Autoencoder (AE) methods. The architecture of XLNet is developed to work seamlessly with the AR objective, including integrating Transformer-XL and the design of the two-stream attention mechanism. Experimental results show that XLNet achieves substantial improvement over previous pre-training methods on various tasks [23]. In this study, we use XLNet to create embeddings and pre-train the XLNet-Base [24] which contains 12-layer, 768-hidden, and 12-heads.

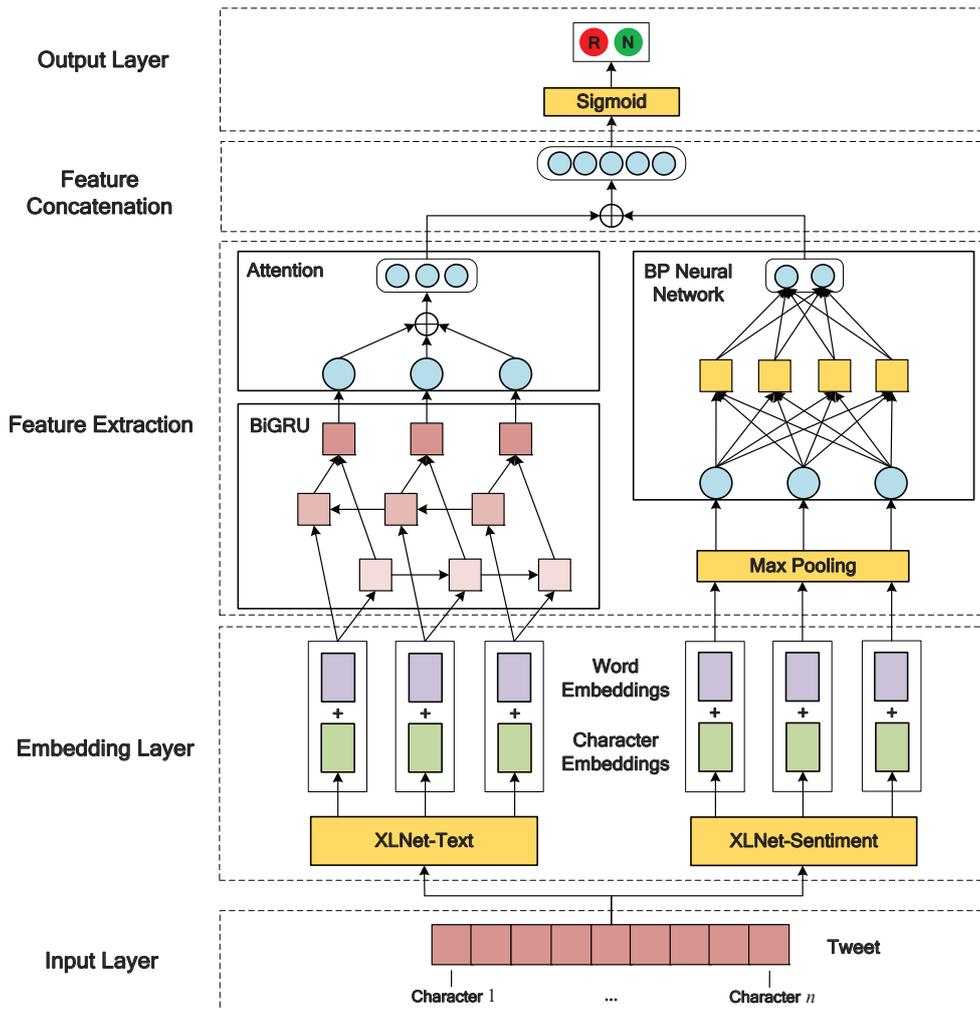


Figure 1. Structure of the XLNet-based BiGRU network with Attention mechanism (XGA) model.

### 3.2.1. Text-based Embeddings

Firstly, in order to capture the features of Cantonese, we construct a multi-domain Cantonese corpus, which includes new articles/blogs, the entities on Encyclopedia of Virtual Communities in Hong Kong (EVCHK) [25], restaurant reviews, forum threads, etc. Then, XLNet-Base is pre-trained on the corpus and fine-tuned using our constructed CR-Dataset. Specifically, the tokens of all the characters in the tweet are fed into the XLNet model and we get 768-dimensional vectors  $C_t = \{C_{t_1}, C_{t_2}, \dots, C_{t_n}\}$ , which are the outputs of the last hidden layer. In order to obtain the words' external contexts and internal structures, we present an addition operation for semantic features between  $C_t = \{C_{t_1}, C_{t_2}, \dots, C_{t_n}\}$  (character embeddings) and Cantonese word embeddings  $W = \{W_1, W_2, \dots, W_n\}$  that provided by fastText [26]. The  $n$  denotes the number of characters in the tweet. Finally, we take the results of the addition operation as text-based embeddings  $E_t = \{E_{t_1}, E_{t_2}, \dots, E_{t_n}\}$  and then feed them into the BiGRU model.  $E_t$  is calculated by

$$E_{t_i} = C_{t_i} + W_i, \tag{1}$$

where  $E_{t_i}$  represents the text-based embedding of  $i$ th character in the tweet,  $C_{t_i}$  represents the character embedding of  $i$ th character in the tweet, and  $W_i$  represents the word embedding of  $i$ th character in the tweet.

### 3.2.2. Sentiment-based Embeddings

Compared to non-rumors, rumors are more inflammatory and deceptive. Therefore, the sentiment polarity of most rumors tends to be negative. For example, a rumor on Twitter is that “顏色水根本唔係水！係化學毒劑，令皮膚很灼熱刺痛！唔好被政府誤導，唔好叫顏色水！請大家以後叫它「化學毒劑」或「化武車」！化學毒劑傷害民眾，殘留毒物污染社區，已經瘋狂失控！” (Color water is not water at all! It is a chemical poison that makes the skin very burning and stinging! Don't be misled by the government and don't call it color water! Please call it “chemical toxic agent” or “chemical weapon vehicle”! This chemical toxic agent hurts people and the residual poisons pollute the community. It is out of control!). The probability that the sentiment polarity of this rumor to be negative is 73%, which is given by the sentiment analysis of Baidu AI [27]. Since the text-based embeddings focus on the semantic features of tweets, it is difficult for them to capture much information about sentiment. So we propose a model to create sentiment embeddings and extract sentiment features of tweets.

In most cases, supervised machine learning approaches are used to train a sentiment classifier with labeled data. But in this research, no data annotation indicating the sentiment polarity of tweets has done on CR-Dataset. Thus, we fine-tune a pre-trained model on a Cantonese dataset with sentiment polarity to solve this problem.

Similar to the work in Section 3.2.1, we first pre-train the XLNet-Base using a multi-domain Cantonese corpus. The pre-trained XLNet is then fine-tuned by the openrice-senti dataset [28], which contains random reviews of restaurants from OpenRice Hong Kong Section [29]. Then the tweet in CR-Dataset is fed into the pre-trained XLNet. We add the outputs of the last hidden layer, which are the character embeddings  $C_s = \{C_{s_1}, C_{s_2}, \dots, C_{s_n}\}$ , with the word embeddings  $W = \{W_1, W_2, \dots, W_n\}$ . The sentiment-based embeddings  $E_s = \{E_{s_1}, E_{s_2}, \dots, E_{s_n}\}$  are calculated by

$$E_{s_i} = C_{s_i} + W_i, \quad (2)$$

where  $E_{s_i}$  represents the sentiment-based embedding of  $i$ th character in the tweet,  $C_{s_i}$  represents the character embedding of  $i$ th character in the tweet, and  $W_i$  represents the word embedding of  $i$ th character in the tweet.

### 3.2.3. Joint Learning of Character and Word Embeddings

Most word embedding methods take a word as a basic unit and learn embeddings according to words' external contexts, ignoring the internal structures of words. However, in Cantonese, a word is usually composed of several characters and contains rich internal information [30]. The semantic meaning of a word is also related to the meanings of its composing characters. In some cases, a single character in Cantonese is very ambiguous and may be composed of multiple words. If a character is used as a semantic unit, it cannot accurately represent the current contextual information. As an example, “鬼唔知咩” means that who doesn't know. “鬼” has different meanings in Cantonese. For instance, it can be used as a metaphor for people with various characteristics. But in this word, it serves as the subject as an interrogative pronoun. This example shows that we cannot use a single character as a semantic unit. So, in this part, we introduce internal character information into word embedding methods to alleviate excessive reliance on external information.

## 3.3. Feature Extraction

In this work, we take the text-based embeddings as the inputs of BiGRU. Then, the attention mechanism is used to focus on the important words in the tweet and output a 150-dimensional vector which indicates the semantic features of the tweet. In addition, we perform the max-pooling step on the sentiment-based embeddings to map the features to a lower-dimensional space. Then, we make use of the Back Propagation (BP) neural network to learn the implicit relationship between features and obtain a 50-dimensional vector which indicates the sentiment features of the tweet.

### 3.3.1. Bidirectional Gated Recurrent Unit

Gated Recurrent Units (GRU) model consists of two gates: update gate and reset gate. The update gate controls whether the status of GRU is updated or how many units are updated. The reset gate determines how much previous information should be ignored. We use BiGRU for two reasons. One is to solve the problem of vanishing gradients. The other is to obtain contextual information.

Specifically, the model feeds the text-based embeddings  $E_t = \{E_{t_1}, E_{t_2}, \dots, E_{t_n}\}$  into the BiGRU network to learn the contextual features of the tweet. The output  $H = \{h_1, h_2, \dots, h_k\}$  is a 150-dimensional vector.  $k$  is the number of hidden units in the network.  $H = \{h_1, h_2, \dots, h_k\}$  is given by

$$\vec{h}_i = GRU(E_{t_i}, \vec{h}_{i-1}), \quad (3)$$

$$\overleftarrow{h}_i = GRU(E_{t_i}, \overleftarrow{h}_{i+1}), \quad (4)$$

$$h_i = \vec{h}_i \oplus \overleftarrow{h}_i, \quad (5)$$

where  $\vec{h}_{i-1}$  is the state generated in the previous step of GRU,  $\overleftarrow{h}_{i+1}$  is the state generated in the next step of GRU, and  $h_i$  is the output of BiGRU. The  $\oplus$  denotes the concatenation of two vectors.

### 3.3.2. Attention

Our model uses the attention mechanism to automatically discover the typical words in the rumor detection and capture the most important semantic information from each tweet. In addition, the input sentence in this research is long. If all the semantic information is represented by an intermediate vector, it would lead to the loss of many details. So it is necessary to introduce the attention mechanism which can give higher weights to the words related to rumors and improve the accuracy of rumor detection.

Specifically, we use the attention mechanism to assign different weights to the outputs of BiGRU  $H = \{h_1, h_2, \dots, h_k\}$  according to their importance. The 150-dimensional vector  $F_t$  that indicates semantic features is described by

$$F_t = Attention(H). \quad (6)$$

### 3.3.3. Back Propagation Neural Network

We add a max pooling layer to obtain new sentiment-based embeddings with smaller dimensions, which are then mapped to a 50-dimensional vector through a BP neural network.

Pooling is a technique of reducing spatial dimensions. It can reduce the number of parameters to learn and the amount of computation performed in the network. The max pooling layer in our model is to take the maximum value of each dimension of the vectors [7]. To be specific, the size of sentiment-based embeddings  $E_s$  is turned from  $13,000 \times 140 \times 768$  to  $13,000 \times 768$  through the max pooling layer. The output of the max pooling layer  $E'_s$  is described by

$$E'_s = Pooling(E_s). \quad (7)$$

Then,  $E'_s$  is mapped to a 50-dimensional vector  $F_s$  through the fully connected layer of the BP neural network. The  $F_s$  indicates the sentiment features of the tweet, which is given by

$$F_s = f(w_s \cdot E'_s + b_s), \quad (8)$$

where  $f()$  is the activation function,  $w_s$  is a weight matrix in the trained detection model, and  $b_s$  is the bias term.

### 3.4. Feature Concatenation

The semantic and sentiment features of the tweet are distinguishing in the task of rumor detection. So we concatenate  $F_t$  and  $F_s$  to obtain a 200-dimensional vector  $F$ , which indicates all the features extracted for Cantonese rumor detection.  $F$  is given by

$$F = F_t \oplus F_s. \quad (9)$$

### 3.5. Output Layer

The vector  $F$  is passed into the *Sigmoid* function to obtain the result of classification, which is given by

$$p = \text{Sigmoid}(F), \quad (10)$$

$$y = \begin{cases} 0, & p \in [0, 0.5) \\ 1, & \text{otherwise} \end{cases}, \quad (11)$$

where  $p$  is the possibility that the tweet is a rumor, and  $p \in [0, 1]$ . The  $y$  is the classification result. In this binary classification task,  $y = 1$  indicates a rumor, and  $y = 0$  indicates a non-rumor.

## 4. Experiments and Evaluation

In this section, we evaluate the performance of the proposed XGA model based on CR-Dataset. All experiments were undertaken on a workstation with two Tesla-V100 32G GPUs. In the experiments, we held out 80% of CR-Dataset for training, 10% for validation, and 10% for testing. The results shown in this section are the average value of each experiment that was repeated ten times independently.

Four metrics are used to evaluate the performance of the embedding layer and the proposed detection model, including *Accuracy*, *Precision*, *Recall*, and *F-score*. The *True Positive (TP)* is the number of rumors that are correctly detected, the *False Negative (FN)* is the number of rumors that are incorrectly detected, the *False Positive (FP)* is the number of non-rumors that are incorrectly detected, and the *True Negative (TN)* is the number of non-rumors that are correctly detected. *Accuracy*, *Precision*, *Recall*, and *F-score* can be computed by

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \quad (12)$$

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (13)$$

$$\text{Recall} = \frac{TP}{TP + FN}, \quad (14)$$

$$F\text{-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (15)$$

In addition, we plot a Receiver Operating Characteristic (ROC) curve based on *True Positive Rate (TPR)* and *False Positive Rate (FPR)* and compute the Area Under Curve (AUC) score to evaluate the performance of our proposed model. *TPR* and *FPR* can be computed by

$$\text{TPR} = \frac{TP}{TP + FN}, \quad (16)$$

$$\text{FPR} = \frac{FP}{FP + TN}. \quad (17)$$

### 4.1. Evaluation of the Embeddings

We use XLNet as the embedding extractor in the XGA model. To evaluate the effectiveness of XLNet, we compare it with the other five popular word embedding models. Note that the structures of

the other parts in the model remain unchanged. The performance of models with different embeddings is shown in Table 1.

**Table 1.** The performance of models with different embeddings.

Embedding	Accuracy	Precision	Recall	F-Score
BERT	0.9119	0.9099	0.9102	0.9101
GPT	0.9099	0.9092	0.9063	0.9076
ELMo	0.9048	0.9048	0.9046	0.9047
fastText	0.9030	0.9031	0.9028	0.9029
Word2vec	0.9079	0.9056	0.9063	0.9060
XLNet	0.9200	0.9224	0.9142	0.9176

In this study, we have pre-trained and fine-tuned the XLNet model on Cantonese data to achieve a better performance of rumor detection. The XLNet used in this experiment is the original XLNet-Base for the fairness of the experiment. As shown in Table 1, XLNet outperforms Bidirectional Encoder Representations from Transformers (BERT) [31], Generative Pre-Training (GPT), and Embeddings from Language Models (ELMo) [32]. This is because XLNet leverages the best of both AR language modeling and AE while avoiding their limitations. Specifically, since an AR language model (e.g., ELMo, GPT) is only trained to encode a uni-directional context, it is not effective at modeling deep bidirectional contexts, which are often required in the downstream language understanding tasks. In comparison, an AE pre-training method (e.g., BERT) is allowed to utilize bidirectional contexts. However, the artificial symbols like [MASK] used by BERT during pre-training are absent from real data at fine-tuning, resulting in a pretrain-finetune discrepancy [23]. In addition, XLNet is better than fastText and Word2vec. This is because XLNet generates contextualized embeddings, which are computed for a word based on its context by pre-trained models, while fastText and Word2vec produce embeddings for each word regardless of its context. Therefore, XLNet achieves the best result among all the models.

#### 4.2. Ablation Study

There are two kinds of features used in our model for rumor detection, i.e., semantic features and sentiment features. In addition, the attention mechanism and word embeddings play an important role in the model. To evaluate the contributions of these significant components to the model, we take turns to exclude them from the model:

- XGA-SF-1: Only the semantic features are used.
- XGA-SF-2: Only the sentiment features are used.
- XG: The attention mechanism is removed.
- XGA-CE: The word embeddings are removed and only the character embeddings are used.
- XGA: Full model.

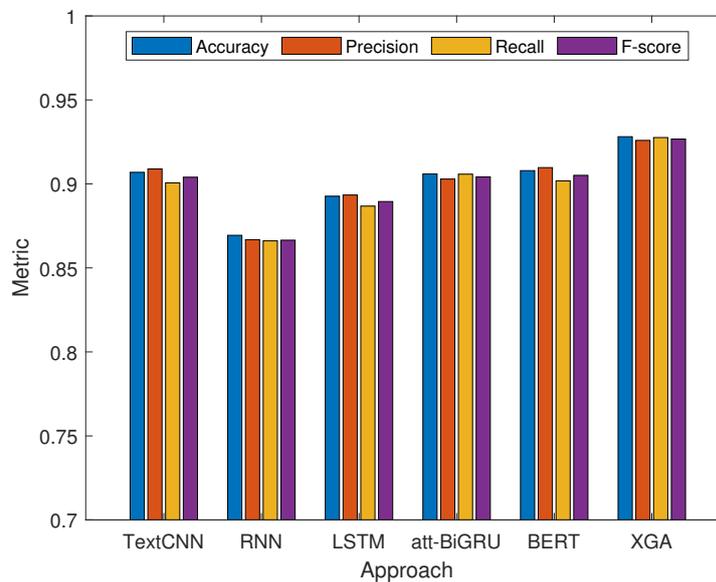
The results of the ablation study are shown in Table 2. We can see that both semantic and sentiment features improve the performance of the model. Moreover, semantic features are more effective than sentiment features in the detection. This is because the sentiment polarity of some tweets is unclear. In addition, since the attention mechanism can give greater weights to typical rumor vocabulary, XGA performs better than XG. What's more, we compare XGA-CE with XGA and find that performing joint learning of character and word embeddings is useful because it can obtain both the words' external contexts and internal information. In conclusion, XGA outperforms all the other models, and all of these components make great contributions to the detection.

**Table 2.** The results of ablation study.

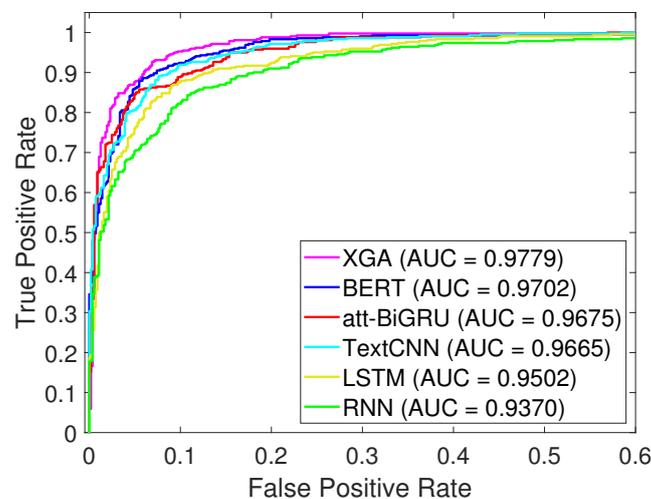
Model	Accuracy	Precision	Recall	F-Score
XGA-SF-1	0.9175	0.9176	0.9178	0.9175
XGA-SF-2	0.8794	0.8820	0.8784	0.8789
XG	0.9157	0.9158	0.9161	0.9157
XGA-CE	0.9030	0.9031	0.9028	0.9029
XGA	0.9281	0.9259	0.9276	0.9267

### 4.3. Evaluation of the XGA Model

We compare the proposed XGA model with other widely used approaches in rumor detection, including TextCNN, RNN, LSTM, att-BiGRU (BiGRU with the attention mechanism), and BERT. We use *Accuracy*, *Precision*, *Recall*, and *F-score* as the evaluation metrics of the detection approaches. The performance of different deep learning approaches and XGA is shown in Figure 2. The ROC curve is shown in Figure 3.



**Figure 2.** The performance of different deep learning approaches and XGA.



**Figure 3.** The ROC curve.

As shown in Figure 2, we can see that the XGA model achieves the F-score of 0.9267 on CR-Dataset and shows the best result in all the four metrics. This is because XLNet uses Transformer-XL as its feature extractor, which has better performance than CNN (used by TextCNN), RNN (used by RNN, LSTM, and att-BiGRU), and Transformer [33] (used by BERT) on semantic feature extraction. In addition, the AUC score shown in Figure 3 proves that att-BiGRU performs better than TextCNN, RNN, and LSTM, which indicates that BiGRU and the attention mechanism have an advantage of extracting features, and that is why we use them in the XGA model to create semantic feature vectors for Cantonese tweets. What's more, the XGA model outperforms att-BiGRU by a margin of 2%, which proves the effectiveness of XLNet and sentiment views involved in our model. In conclusion, compared with other deep learning approaches, the proposed XGA model using XLNet, BiGRU, and the attention mechanism is more effective in Cantonese rumor detection. In the experiment, among the test set which includes 1300 tweets, non-rumors are misclassified 55 times, and rumors 41 times. In some cases, a non-rumor is posted to refute a certain rumor and often contains reasoning or turns. For example, a non-rumor may first explain an existing rumor, and use examples or reasoning to prove that the rumor is false in the subsequent text. So, the final detection result of the non-rumor will be affected by the rumor with negative emotional tendencies. That is the reason why some non-rumors cannot be detected correctly. Moreover, the sentiment polarity of some rumors is unclear, which results in the misclassification of rumors. In the future, to reduce the misclassification, we can optimize our model to make it learn the mutual negation between paragraphs in the tweet.

## 5. Conclusions and Future Work

In this paper, we have proposed a novel hybrid model called XGA for detecting Cantonese rumors, which takes advantage of XLNet, BiGRU, and the attention mechanism. To be specific, we extracted both semantic and sentiment features for the detection. The XLNet, which was pre-trained and fine-tuned on Cantonese data, was used to produce text-based and sentiment-based embeddings. In addition, we combined the character embeddings extracted by XLNet with Cantonese word embeddings to learn the words' external contexts and internal structures. Furthermore, we made use of BiGRU and the attention mechanism to obtain the important semantic features, which were then concatenated with sentiment features to get the final classification results. We performed two experiments to evaluate the effectiveness of our model and came to the following conclusions: the XLNet performed better than other word embedding models, and the XGA model we designed achieved the F-score of 0.9267 in Cantonese rumor detection and outperformed other widely used detection models in all metrics.

In the future, we plan to conduct further research on Cantonese word segmentation to improve the performance of word embeddings. In addition, we will try to discover more effective features for the Cantonese rumor detection and make use of them in our model.

**Author Contributions:** Conceptualization, H.W., X.C., L.K., H.S. and Z.L.; methodology, H.W., X.C., and L.K.; validation, X.C. and L.K.; formal analysis, X.C. and L.K.; investigation, X.C. and L.K.; data curation, X.C., L.K., Z.L. and H.S.; writing—original draft preparation, X.C. and L.K.; writing—review and editing, X.C., L.K., H.W., Z.L. and H.S.; supervision, H.W.; project administration, X.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Acknowledgments:** This work was completed under the guidance of Haizhou Wang of Sichuan University, China.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Liang, G.; He, W.; Xu, C.; Chen, L.; Zeng, J. Rumor identification in microblogging systems based on users' behavior. *IEEE Trans. Comput. Soc. Syst.* **2015**, *2*, 99–108. [CrossRef]
2. Snopes. Available online: <https://www.snopes.com> (accessed on 28 August 2020).
3. PolitiFact. Available online: <https://www.politifact.com> (accessed on 28 August 2020).

4. 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, New York, NY, USA, 9–15 July 2016; pp. 3818–3824.
5. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. CSI: A hybrid deep model for fake news detection. In Proceedings of the 26th ACM Conference on Information and Knowledge Management, Singapore, 6–10 November 2017; pp. 797–806.
6. Liu, Y.; Wu, Y.F.B. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In Proceedings of the 32nd AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; pp. 354–361.
7. 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; pp. 1980–1989.
8. Bondielli, A.; Marcelloni, F. A survey on fake news and rumour detection techniques. *Inf. Sci.* **2019**, *497*, 38–55. [[CrossRef](#)]
9. Meel, P.; Vishwakarma, D.K. Fake news, rumor, information pollution in social media and web: A contemporary survey of state-of-the-arts, challenges and opportunities. *Expert Syst. Appl.* **2020**, *153*, 112986. [[CrossRef](#)]
10. 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*, 1–36. [[CrossRef](#)]
11. Chen, T.; Li, X.; Yin, H.; Zhang, J. Call attention to rumors: Deep attention based recurrent neural networks for early rumor detection. In Proceedings of the 22nd Pacific-Asia Conference on Knowledge Discovery and Data Mining, Melbourne, Australia, 3–6 June 2018; pp. 40–52.
12. Ma, J.; Gao, W.; Wong, K.F. Detect rumors on twitter by promoting information campaigns with generative adversarial learning. In Proceedings of the 28th The World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; pp. 3049–3055.
13. Kochkina, E.; Liakata, M.; Zubiaga, A. All-in-one: Multi-task Learning for Rumour Verification. In Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, NM, USA, 20–26 August 2018; pp. 3402–3413.
14. Kumar, S.; Carley, K.M. Tree lstms with convolution units to predict stance and rumor veracity in social media conversations. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 5047–5058.
15. Yu, F.; Liu, Q.; Wu, S.; Wang, L.; Tan, T. A convolutional approach for misinformation identification. In Proceedings of the 26th International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017; pp. 3901–3907.
16. Qian, F.; Gong, C.; Sharma, K.; Liu, Y. Neural user response generator: Fake news detection with collective user intelligence. In Proceedings of the 27th International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13–19 July 2018; pp. 3834–3840.
17. 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.* **2019**, *1*. [[CrossRef](#)]
18. 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.
19. Ma, J.; Gao, W.; Joty, S.; Wong, K.F. An Attention-based Rumor Detection Model with Tree-structured Recursive Neural Networks. *ACM Trans. Intell. Syst. Technol.* **2020**, *11*, 42. [[CrossRef](#)]
20. 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. In Proceedings of the 34th AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 549–556.
21. Dashtipour, K.; Gogate, M.; Li, J.; Jiang, F.; Kong, B.; Hussain, A. A hybrid Persian sentiment analysis framework: Integrating dependency grammar based rules and deep neural networks. *Neurocomputing* **2020**, *380*, 1–10. [[CrossRef](#)]
22. Cantonese Rumor Dataset. Available online: <https://github.com/cxyccc/CR-Dataset> (accessed on 28 August 2020).

23. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.R.; Le, Q.V. Xlnet: Generalized autoregressive pretraining for language understanding. In Proceedings of the 33rd Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 5753–5763.
24. XLNet-Base. Available online: <https://github.com/zihangdai/xlnet> (accessed on 28 August 2020).
25. Encyclopedia of Virtual Communities in Hong Kong. Available online: <https://evchk.wikia.org/zh/wiki/> (accessed on 28 August 2020).
26. FastText Pre-trained Vectors. Available online: <https://fasttext.cc/docs/en/pretrained-vectors> (accessed on 28 August 2020).
27. Sentiment Analysis of Baidu AI. Available online: [https://ai.baidu.com/tech/nlp\\_apply/sentiment\\_classify](https://ai.baidu.com/tech/nlp_apply/sentiment_classify) (accessed on 28 August 2020).
28. Openrice-senti Dataset. Available online: <https://github.com/toastynews/openrice-senti> (accessed on 28 August 2020).
29. OpenRice Hong Kong Section. Available online: <https://www.openrice.com/zh/hongkong> (accessed on 28 August 2020).
30. Chen, X.; Xu, L.; Liu, Z.; Sun, M.; Luan, H. Joint learning of character and word embeddings. In Proceedings of the 24th International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 July 2015; pp. 1236–1242.
31. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 17th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
32. Peters, M.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep Contextualized Word Representations. In Proceedings of the 16th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, LA, USA, 1–6 June 2018; pp. 2227–2237.
33. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA, 19 May 2017; pp. 5998–6008.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).