Rumor Detection on Social Media: A Multi-View Model using Self-Attention Mechanism

Yue Geng[1,2], Zheng Lin[1*], Peng Fu[1], and Weiping Wang[1]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
{gengyue, linzheng, fupeng, wangweiping}@iie.ac.cn

Abstract. With the unprecedented prevalence of social media, rumor detection has become increasingly important since it can prevent misinformation from spreading in public. Traditional approaches extract features from the source tweet, the replies, the user profiles as well as the propagation path of a rumor event. However, these approaches do not take the sentiment view of the users into account. The conflicting affirmative or denial stances of users can provide crucial clues for rumor detection. Besides, the existing work attaches the same importance to all the words in the source tweet, but actually, these words are not equally informative. To address these problems, we propose a simple but effective multi-view deep learning model that is supposed to excavate stances of users and assign weights for different words. Experimental results on a social-media based dataset reveal that the multi-view model we proposed is useful, and achieves the state-of-the-art performance measuring the accuracy of automatic rumor detection. Our three-view model achieves 95.6% accuracy and our four-view model using BERT as a view also reaches an improvement of detection accuracy.

Keywords: Rumor detection · Multi-view model · Self-Attention · Deep learning.

1 Introduction

Nowadays, social media enable not only journalists but also ordinary individuals to post ongoing events. As social media provide citizens with an ideal platform to stay abreast of momentous events, it is also eligible for broadcasting rumors. Rumors that are ultimately proven false often have damaging consequences in view of the fact that they negatively impact citizens' life and sometimes even trigger public panic. For instance, a rumor claiming that iodized salt could prevent radiation was posted in 2011 and a great number of citizens stripped supermarkets of salt in the belief that it could ward off radiation poisoning [6]. This piece of fake news caused panic in population as well as a huge market disorder. Therefore, it is crucial to identify rumors in social media where large amounts of information

* Corresponding author

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

are easily spread. This emphasizes the need for studies that can assist in analyzing the veracity of news. Rumors can give rise to shock, suspicion or protest in public since misinformation affects individuals' perception of events and causes harmful consequences which tend to be more severe over time.

To alleviate these problems, studies have been conducted from different perspectives ranging from psycholinguistic analysis to deep learning techniques. Early studies extracted groups of related features (i.e., message, topic) and built a machine learning classifier to evaluate the credibility of social media posts [16]. One of the drawbacks of this kind of method is that hand-crafted features hardly explore the inner relationship among replies. Recently, Ma et al. [12] exploited recurrent neural networks (RNN) to represent the content of the source tweet and its replies/retweets. RNNs automatically learn both temporal and textual features and thus yield outstanding performance. This work detected rumor events mainly based on contents whereas the stances of users were not concentrated on. Since those users who read the rumors may have common sense, and they may share opinions or raise questions on suspicious posts, we introduce a new sentiment view for rumor detection. Specifically, since source tweet and replies are proven useful in previous work, a supplementary sentiment view is adopted, and then the multi-view model is constructed. Besides, the previous work tokenizes the posts and equally treat each word while we train a self-attention layer which pays more attention to significant words in the source tweet. We train and test our model based on a Weibo dataset[3]. This dataset incorporates 4664 events and the posts in the event are sorted by time. In each event, a source tweet is associated with a number of replies, retweets, and user profiles. We utilize the dataset to understand how the lexical content and the users' reactions are related to its veracity. Our work indicates that the content of all the posts and the users' sentiment are capable of better exploiting representations of rumors. Our research also reveals that GRU with self-attention mechanism [19] can provide strong assistance for social-media-based rumor detection. The source code is available at GitHub[4].

The main contributions of our research include:

- We develop a multi-view network which analyzes features related to a specific event in social media. This model enables deep neural networks to learn representations containing adequate information from three different perspectives, including the source tweets, replies/retweets, and the latent sentiment semantics. All of the views we proposed are useful for rumor detection based on experimental results.
- We also apply the Gated Recurrent Unit (GRU) with self-attention mechanism to better capture the features of the content as well as the propagation path of a certain event and automatically assign a weight for each reply/retweet corresponding to its significance.
- Our model demonstrates the state-of-the-art performance of detecting rumors in social media on Weibo dataset. Our three-view model outperforms

[3] `http://alt.qcri.org/~wgao/data/rumdect.zip`
[4] `https://github.com/crystalyue/multi-view-rumor-detection`

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

baseline PPC_RNN+CNN [10] by 3.5%. We also test the performance of the Bidirectional Encoder Representations from Transformers (BERT) model [5] as the fourth view, and the combination of our proposed model with BERT model can achieve an even better result.

The rest of this paper is organized as follows. We begin with an overview of related work in Section 2. Section 3 presents the rumor detection task and a detailed description of our multi-view system including the preprocessing methods and the feature sets. Section 4 introduces the datasets used in this paper and provides the experimental settings. We also analyze the detection performance of our model. The purpose of the evaluation experiments is to compare the predictive capacity with that of the prevailing methods. Besides, we conduct experiments on assessing the contributions of different parts of our model to the overall performance. Finally, we conclude and present directions for future work in Section 5.

2 Related Work

Related work on rumor detection can be roughly classified into four categories, content-based, knowledge-based, propagation-path-based, and hybrid methods.

Qian et al. [17] introduced a Two-Level Convolutional Neural Network with User Response Generator (TCNN-URG) where Two-Level Convolutional Neural Network (TCNN) captures underlying semantic information at both word and sentence levels. User Response Generator (URG) is based on Conditional Variational Autoencoder (CVAE) which generates user responses to new articles with the assistance of historical user responses. Sarkar et al. [4] proposed a different idea which is to build a hierarchical neural network architecture. First, they took a sequence of weighted average word embeddings as inputs to generate a sentence embedding. Second, they created a document embedding taking the sentence embeddings as inputs. The resulting document embeddings contain semantic information at both sentence and document levels.

In addition, there are other groups of approaches. Knowledge-based rumor detection methods mainly focus on information retrieval or knowledge graph. By extracting basic elements of the document and searching them from websites, Wu et al. measure the quality of the query results. The results are accumulated to obtain a final score for a given document [21]. The other possible means is to build a Wikipedia Knowledge Graph and evaluate the veracity of news by calculating the truth value of the shortest path between entities in the knowledge graph. A subject-predicate-object statement's veracity amounts to both the path length between the two target entities and the generality of the entities [3].

Studies also found that temporal features could strengthen the predictive power of models. Kwon et al. [9] proposed a time-series-fitting model representing a rumor's spreading pattern. Ma et al. extracted more time-sensitive features and explored how they vary in time [13]. Subsequently, Jin et al. discovered conflicting viewpoints in tweets by constructing a credibility propagation network of tweets. Based on a topic model method, the credibility propagation finally

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

generates a credit score for each piece of news [7]. Kernel-based methods are also capable of automatically modeling the propagation path of an event. Propagation trees giving clues for how tweets are transmitted in their life span. Ma et al. then identified rumors by comparing the similarities between different tweets' propagation tree structures [14].

In the meanwhile, researchers introduced hybrid deep learning models which drastically improve the accuracy of rumor detection. Apart from linguistic features, user profiles including party affiliation, speaker title, location and credit history can also be used as additional information. Long et al. [11] included user profile information in attention layers. Volkova et al. showed that a joint learning neural network model based on social network interactions and news contents advanced lexical models since syntax and grammar features did not make any contribution to evaluate the veracity of rumors [20]. Liu et al. modeled the propagation path using ensemble learning and encoded eight kinds of social-media-based user characteristics. They then built a classifier that incorporates both RNN and CNN to utilize textual contents as well as user characteristics along the propagation path [10]. Moreover, Ma et al. proposed a bottom-up and a top-down tree-structured neural network both of which naturally model the propagation paths of a rumor [15].

However, existing network-based methods did not focus on the crucial opinions in the replies as well as the different extent of information that different words in the source post can provide. In this work, we use a multi-view representation model to exploit contents, replies and the supporting and opposing sentiment to improve the performance of rumor detection.

3 Method

In this section, we present the details of our proposed model for classifying rumor events. First, we describe the overall structure of our model and introduce a method that assigns different weights for each word in the source tweet. Then, we describe each part of our model and explain how the sentiment view is constructed.

3.1 Problem Statement

Let $E = \{e_1, e_2, ..., e_n\}$ be an event where e_1 is the source tweet and $\{e_2, ..., e_n\}$ are the posts related to the source tweet. Each event is associated with a label L indicating whether the source tweet in this event is a rumor or not. Note that $L = 0$ denotes the event is a non-rumor while $L = 1$ denotes the event is a rumor. Our rumor detection task can be defined as automatically distinguishing the veracity of an event given its corresponding label L.

3.2 The Proposed Model

Overview. We propose a multi-view neural network model to classify Weibo posts into two categories—rumor and non-rumor. The architecture was present-

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

ed in Figure 1. Three views are incorporated in our model which are named as content view, reply view and sentiment view respectively. All the source tweets and the reply/retweet sentences are tokenized using *Jieba* tools[5]. We initialize our embedding layer with pre-trained 200-dimensional word embeddings for Chinese words and phrases following the setup as in [18]. We implement and train the proposed model using *PyTorch*[6].

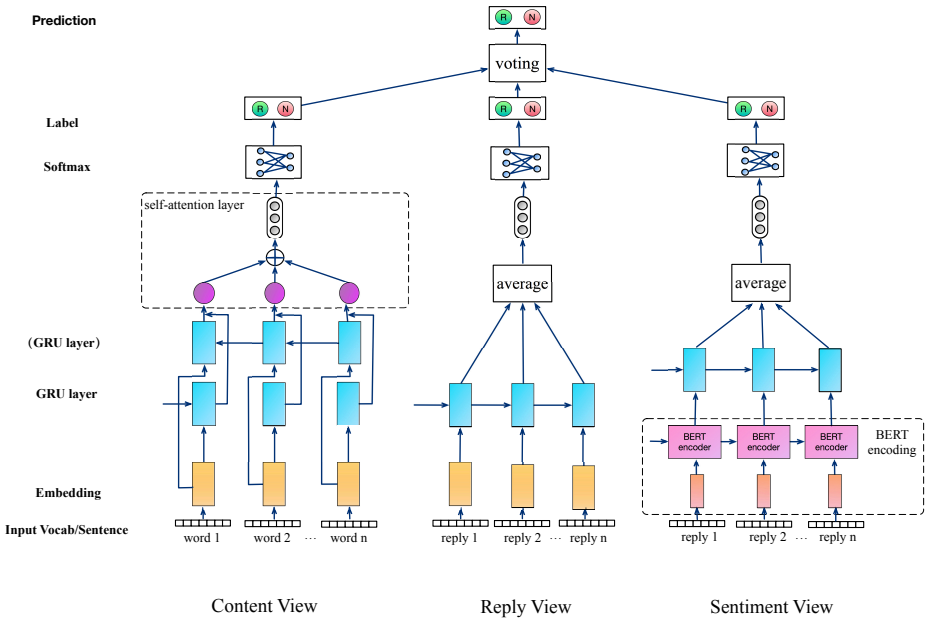


Fig. 1. The framework of the proposed multi-view model. The content view combines bidirectional GRU and a self-attention layer to evaluate the veracity of the source tweet in an event. The reply view generates representations for each reply/retweet in the event through the GRU layer. The sentiment view extracts sentiment embeddings via fine-tuned BERT encoders. The vote layer integrates the results of three views and finally predicts a label by majority voting.

The content sub-network consists of an embedding layer and a GRU network with a self-attention layer. For the content view, GRU takes each word's embedding in the source post as input. In the embedding layer, the source tweet S_n with a length l is represented as a vector $[S_1, S_2, ..., S_l]$ where Si is the word embedding of the l-th word. The network is followed by a self-attention layer for the reason that not all the words in a given post have the same significance. Thus, we calculate the similarities among the words in the post and calculate the weight for each word. Since it is unlikely to accurately distinguish rumors from

[5] `https://pypi.org/project/jieba/`
[6] `https://pytorch.org/`

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

true news only depending on its source tweet, we proposed other views providing more information to assist in promoting the performance of our model. For the reply view, we average the word embeddings to produce a post embedding which enables GRU to receive a reply/retweet at a time. GRU is used for excavating the lexical features and temporal features of events. In addition to content and reply views, we also generate sentiment embedding for each post in the event with the assistance of a fine-tuned BERT model and then send the embeddings to GRU. Each view is followed by a softmax layer which predicts a result, i.e., 1 for rumor and 0 for non-rumor. Take the three-view model as an example, the results of the three views are combined as a majority vote classifier. If there are two or more views generate the same result, then that result will be regarded as the final prediction of our model. The voting procedure can be described using the following equation:

$$Predict_{output}(event) = u(\sum_{v \in V} Predict_v(event) - \frac{|V|}{2}), \\ V \subseteq \{content, reply, sentiment\}, V \neq \varnothing \tag{1}$$

$$u(x) = \begin{cases} 1 & x \geq 0, \\ 0 & x < 0. \end{cases} \tag{2}$$

where $Predict_{output}(event)$ indicates the prediction label of the current processing event based on each view's output. V is a non-empty subset of {content, reply, sentiment}. The prediction of each view can be 0 or 1 and if the sum of all the predictions is greater than the threshold $\frac{|V|}{2}$, then we consider the current event as a rumor event.

Propagation Path Modelling via GRU. In terms of framework, we employ the gated recurrent units in order to better capture lexical and temporal information since the standard recurrent neural network is a biased model, where the earlier inputs are more likely to be abandoned during training. GRU receives each word embedding in content view and receives document embedding in reply view, and post-based sentiment embedding is used in sentiment view. GRU [2] is designed to dynamically remember and forget the information flow. There are two types of gates controlling how information is updated, i.e., reset gate and update gate. A single layer GRU accepts input vectors $< x_1, x_2, ..., x_N >$, computes the corresponding hidden states $< h_1, h_2, ..., h_N >$. Specifically, let $\odot$ denote the element-wise product of two vectors, the single layer GRU computes the hidden state h at time t and the corresponding output as:

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \widetilde{h_t} \tag{3}$$

$$\widetilde{h_t} = tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \tag{4}$$

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{5}$$

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{6}$$

where r_t and z_t are reset and update gates respectively. h_t is the hidden state of GRU and $\widetilde{h_t}$ is the candidate output. $\sigma(\cdot)$ are element-wise sigmoid functions. Mean square error (MSE) is used as the loss function for model training. We choose Adam algorithm [8] for updating network parameters.

Self-Attention-Based Content Representation. Self-attention [19] is a special kind of attention mechanism where the query Q is replaced by an embedding x_j from the source input itself. d_k is the dimension of Q, K, V. Specifically, the query Q, key K and value V are the same in this scenario. Self-attention computes the attention between elements at different positions in the sequence. For each embedding pair x_i and x_j, we calculate the Scaled Dot Product as the attention weight. Then we adopt a softmax function to normalize these weights. Finally, we weighted sum the weights and the corresponding value to obtain the final attention scores.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{7}$$

The adoption of the self-attention mechanism is based on the assumption that not all the posts make equal sense. In the self-attention module, query Q, key K and value V represent the concatenation of each output of the hidden units in GRU. Hence, the resulting probabilities can be regarded as the weight of each post. In this way, the significance of each post can be automatically reflected by its weight and the weighted sum of all the posts encodes a better representation for the sentiment view. Moreover, it requires a small number of parameters and has a very fast computation speed. The Scaled Dot-Product Attention we adopt is presented in Figure 2.

Sentiment View. One of the main challenges of constructing sentiment view is to obtain sentiment embeddings of each post. The usual practice is to supervised train a sentiment classifier with annotated labels. However, in this task, there are no sentiment labels indicating the sentiment polarity of the posts. To overcome this obstacle, it is reasonable to fine-tune a pre-trained sentiment classifier and take the output of its hidden layers as sentiment embeddings. Specifically, we employ a pre-trained Bidirectional Encoder Representations from Transformers (BERT) model [5] which advances the state-of-the-art model in eleven Natural Language Processing (NLP) tasks. In the original paper which proposes BERT, the authors report the experimental results on two model sizes: $\text{BERT}_{\text{BASE}}$ with 12 transformer blocks and 768 hidden size, and $\text{BERT}_{\text{LARGE}}$ with 24 transformer blocks and 1024 hidden size. In our paper, we adopt the first model

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

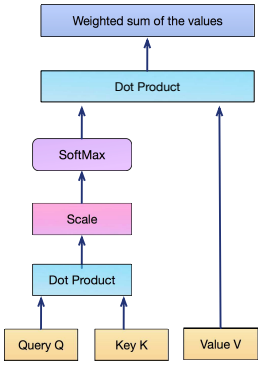


Fig. 2. Scaled Dot-Product Attention adopted in our model.

since $\text{BERT}_{\text{BASE}}$ can already achieve an outstanding result and it is costly to train $\text{BERT}_{\text{LARGE}}$. The architecture of the $\text{BERT}_{\text{BASE}}$ model in our paper is the same as that in the original paper. The $\text{BERT}_{\text{BASE}}$ model is fine-tuned by Weibo sentiment corpus[7] for better excavating the sentiment features of Weibo contents. All the sentences in the posts are fed into the $\text{BERT}_{\text{BASE}}$ model and we take the outputs of the last hidden layer as our sentiment embeddings. The 768-dimensional vectors are then used as the inputs of GRU sentiment view. Figure 3 shows an example of $\text{BERT}_{\text{BASE}}$ model. Since our reply view mainly focuses on contents, it is unlikely for reply view to capture much sentiment information. As a result, we adopt the third view which specifically extract sentiment embeddings. In the sentiment view, we adopt the fine-tuned $\text{BERT}_{\text{BASE}}$ as a sentiment extractor which specifically captures sentiment features. Since $\text{BERT}_{\text{BASE}}$ is a strong state-of-the-art model which outperforms other models in so many NLP tasks, we would like to see if our proposed model can improve its performance. We first use $\text{BERT}_{\text{BASE}}$ without fine-tuning as a single view to test its performance and then combine our 3-view model with the $\text{BERT}_{\text{BASE}}$ model to test the 4-view models performance. Results indicate that this view is not only able to help detect rumor by itself but also enables our 3-view and 4-view model to improve detection accuracy.

4 Experiment

4.1 Experimental Settings

In the experiment, we empirically set the size of hidden units as 300 and the maximum training epoch as 200. The training process finishes when the number of training epoch meets the restriction or the validation loss converges. The input dimension of content/reply view is 200. For Sentiment view, the output dimension of Bert is 768 as the configuration of the pre-trained model and the

[7] `https://github.com/baidu/Senta`

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

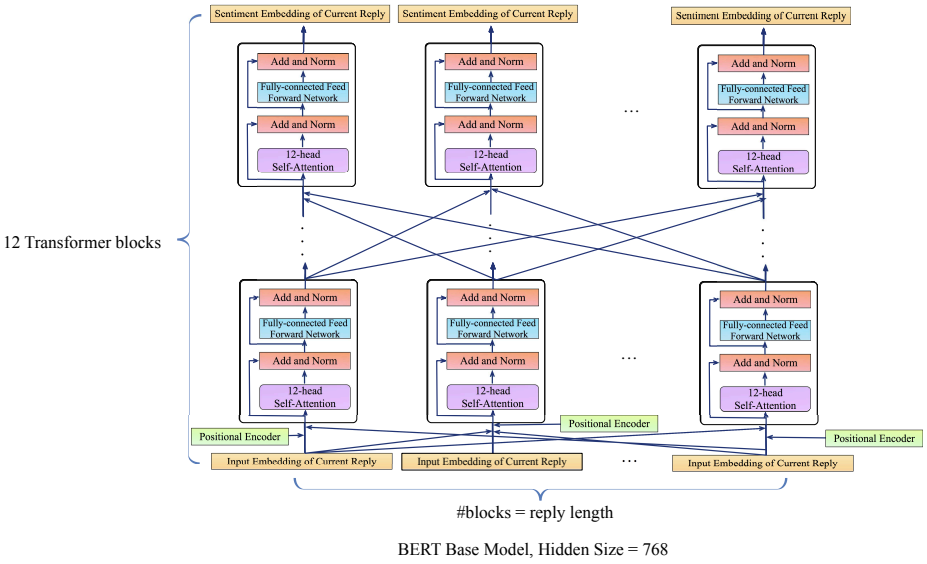


Fig. 3. A sample of BERT model with 12 Transformer blocks.

GRU input embedding dimension is also 768. The output vector size of all the views is set to 300. Batch size is set to 1 and the learning rate is 0.001. We restrict the number of replies in an event to 4096.

4.2 Dataset

We conduct experiments on an available public dataset: Weibo [12]. Posts including a source tweet and its relevant replies/retweets form an event. The dataset is comprised of 4664 events with 2,313 rumors and 2,351 non-rumors. In the same event, posts are sorted by published time and hence a propagation path is naturally constructed. Propagation path represents the extent to which each event is retweeted. Our dataset contains binary labels, i.e., rumor and non-rumor. Table 1 summarizes the statistics of the dataset. To get an overall comparison, we divide our dataset into three subsets by strictly following the same partition configuration as the previous papers. The validation set incorporates 10% of the total events. The remaining rumor events are split in a ratio of 3:1 and are used for model training and testing respectively.

4.3 Baseline Models

We carefully select a series of previous work on rumor classification as baselines, some of which are classical and others are state-of-the-art:

– DTC [1] A decision-tree-based classifier that utilizes a series of hand-crafted features.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

Table 1. Statistics of the Weibo dataset

Statistic	Weibo
# Events	4664
# Rumors	2313
# Non-rumors	2351
# Users	2,746,818
# Posts	3,805,656
Avg. #of posts/event	816

- SVM-RBF [22] An SVM classifier with Radial Basis Function (RBF) kernel that also utilizes hand-crafted features.
- RFC [9] A random forest classifier that fit the utilizes user, linguistic and structure characteristics.
- SVM-TS [13] A linear SVM model that utilizes time-series to model how each kind of features vary in time.
- DT-Rank [23] A ranking method based on the decision tree. Searching for enquiry phrases and ranking the clustered results enable this method to detect rumors.
- GRU-RNN [12] An RNN-based model that learns long-distance dependencies among different time steps, which utilizes more information from user comments.
- PPC_RNN [10] A time series classifier that incorporates recurrent neural networks which combine tweet texts and the user characteristics along the propagation path to detect rumors.
- PPC_CNN [10] A time series classifier that incorporates convolutional neural networks which combine tweet texts and the user characteristics along the propagation path to detect rumors.
- PPC_RNN+CNN [10] A classifier that utilizes RNN and CNN to respectively represent the propagation path, and integrate two paths to detect rumors at the early stage of propagation.

4.4 Results and Discussion

This section presents the experimental results that demonstrate the state-of-the-art performance of our rumor detecting model. Table 2 shows the experimental results of our proposed model and that of baseline models. The three-view model achieves 95.6% accuracy on Weibo dataset. The baseline models listed in the table are carefully selected. So they are representative and classical methods for classification tasks. There are also three recently proposed state-of-the-art models that already achieve a great result, while our proposed model outperforms these baseline models. The reason why our model outperforms PPC_RNN+CNN is that our model introduces sentiment information. Besides, PPC_RNN+CNN adopts ensemble learning with CNNs and RNNs learning features, so that it's time-consuming. It is clear that our model achieves state-of-the-art performance

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

based on all the evaluation indicators, including the overall accuracy, and the precision, recall, F1 score for rumor and non-rumor classes.

Table 2. Fake news detection results on Weibo dataset

Method	Class	Acc.	Prec.	Recall	F1
DTC	R	0.831	0.847	0.815	0.831
	N		0.815	0.847	0.830
SVM-RBF	R	0.818	0.822	0.812	0.817
	N		0.815	0.824	0.819
RFC	R	0.849	0.786	0.959	0.864
	N		0.947	0.739	0.830
SVM-TS	R	0.857	0.839	0.885	0.861
	N		0.878	0.830	0.857
DT-Rank	R	0.732	0.738	0.715	0.726
	N		0.726	0.749	0.737
GRU-RNN	R	0.910	0.876	0.956	0.914
	N		0.952	0.864	0.906
PPC_RNN	R	0.912	0.878	0.958	0.916
	N		0.944	0.866	0.908
PPC_CNN	R	0.919	0.899	0.958	0.922
	N		0.946	0.880	0.916
PPC_RNN+CNN	R	0.921	0.896	0.962	0.923
	N		0.949	0.889	0.918
Content+Reply+Sentiment	R	**0.956**	0.944	**0.966**	0.955
	N		**0.968**	0.947	**0.957**

According to the results in Table 3, we can find that models with two views yield better accuracy than the model with one view. For the models with a different number of views, the superiority of the 3-view model with a voting mechanism is explicit. This result reveals that the selected views make considerable sense and they can capture more useful information from different perspectives. Besides, other kinds of views can also be used in our model. As it was mentioned before, we adopted the pre-trained BERT model to construct a sentiment view. We are also interested in how accurate BERT can achieve in this task. Thus, we combine all the posts in the event as input and train $\text{BERT}_{\text{BASE}}$ model using veracity labels. Obviously, BERT reaches a very great result. We combine four views together, i.e., content, reply, sentiment, Bert using vote classification and assign weights for each view, and we are happy to find that the combined model achieves an even better result on this dataset. This implies that our proposed views do make contributions in the task and can improve BERT's prediction performance.

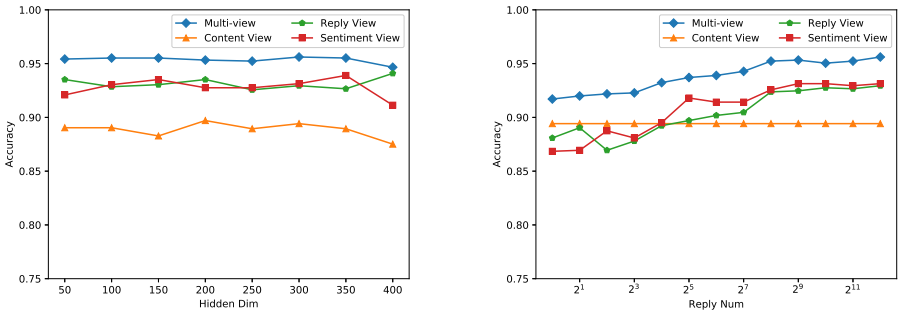
Figure 4 plots the relevance between hyperparameters and detection accuracy. It is clear that when the dimension of the hidden layer is 300, the model performs better than that using an other hidden size. Besides, the model's detection accuracy steadily ascends when the number of replies is increasing. This

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

Table 3. Control Experiment on different views of our model

View	Class	Acc.	Prec.	Recall	F1
Content	R	0.894	0.890	0.888	0.889
	N		0.898	0.899	0.899
Reply	R	0.929	0.908	0.948	0.928
	N		0.950	0.912	0.931
Sentiment	R	0.931	0.935	0.920	0.928
	N		0.928	0.941	0.935
Content+Reply	R	0.953	0.956	0.946	0.951
	N		0.951	0.960	0.955
Content+Sentiment	R	0.938	0.916	0.958	0.937
	N		0.960	0.920	0.939
Content+Reply+Sentiment	R	0.956	0.944	0.966	0.955
	N		0.968	0.947	0.957
Bert	R	0.961	0.941	0.980	0.960
	N		0.981	0.943	0.962
Content+Reply+Sentiment+Bert	R	**0.965**	0.946	**0.982**	0.964
	N		**0.983**	0.949	**0.966**

implies a larger number of replies provides more adequate information thus yield better performance.



(a) Hidden Dim (b) Reply Num

Fig. 4. Rumor detection accuracy when different hyperparameter values are taken.

The overall experiments demonstrate that each view can learn event representations from different perspectives and the fine-tuned pre-trained $\text{BERT}_{\text{BASE}}$ model is able to capture sentiment information expressed in replies and retweets, which can be utilized to guide the prediction for rumor events.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

5 Conclusion and Future Work

In this paper, we provide insights into detecting real-world rumors. We created deep neural networks for automatically predicting the veracity of a rumor and using sentiment embeddings to help better distinguish rumors from true news. Our model achieves higher accuracy than existing baseline models in the task of rumor detection on Weibo dataset. The multi-view model we proposed comprehensively consider source tweet, reply and sentiment information. Despite that our model has a simple structure, it can be a hard-to-beat baseline since the model has already achieved 96.5% accuracy in the defined task. Since our model is generalizable and robust, other insightful views may also be added into it. In addition, we introduced the BERT model into our view, which assists in improving the final detection performance. In the future, we would like to exploit other views and build a more efficient model which has a faster speed but still demonstrates promising results.

Acknowledgments. This work was supported by the National Key Research and Development Program of China under Grant No. 2016YFB1000604.

References

1. Castillo, C., Mendoza, M., Poblete, B.: Information credibility on twitter. In: Proceedings of the 20th international conference on World wide web. pp. 675–684. ACM (2011)
2. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014)
3. Ciampaglia, G.L., Shiralkar, P., Rocha, L.M., Bollen, J., Menczer, F., Flammini, A.: Computational fact checking from knowledge networks. PloS one **10**(6), e0128193 (2015)
4. De Sarkar, S., Yang, F., Mukherjee, A.: Attending sentences to detect satirical fake news. In: Proceedings of the 27th International Conference on Computational Linguistics. pp. 3371–3380 (2018)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
6. Guardian, T.: (2011), chinese panic-buy salt over Japan nuclear threat:`https://www.theguardian.com/world/2011/mar/17/chinese-panic-buy-salt-japan`
7. Jin, Z., Cao, J., Zhang, Y., Luo, J.: News verification by exploiting conflicting social viewpoints in microblogs. In: AAAI. pp. 2972–2978 (2016)
8. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
9. Kwon, S., Cha, M., Jung, K., Chen, W., et al.: Prominent features of rumor propagation in online social media. In: International Conference on Data Mining. IEEE (2013)
10. Liu, Y., Wu, Y.f.B.: Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In: AAAI (2018)

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25

11. Long, Y., Lu, Q., Xiang, R., Li, M., Huang, C.R.: Fake news detection through multi-perspective speaker profiles. In: Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers). vol. 2, pp. 252–256 (2017)
12. 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: IJCAI. pp. 3818–3824 (2016)
13. 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 24th ACM International on Conference on Information and Knowledge Management. pp. 1751–1754. ACM (2015)
14. 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 (Volume 1: Long Papers). vol. 1, pp. 708–717 (2017)
15. 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 (Volume 1: Long Papers). vol. 1, pp. 1980–1989 (2018)
16. Mendoza, M., Poblete, B., Castillo, C.: Twitter under crisis: Can we trust what we rt? In: Proceedings of the first workshop on social media analytics. pp. 71–79. ACM (2010)
17. Qian, F., Gong, C., Sharma, K., Liu, Y.: Neural user response generator: Fake news detection with collective user intelligence. In: IJCAI. vol. 3834, p. 3840 (2018)
18. Song, Y., Shi, S., Li, J., Zhang, H.: Directional skip-gram: Explicitly distinguishing left and right context for word embeddings. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers). vol. 2, pp. 175–180 (2018)
19. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems. pp. 5998–6008 (2017)
20. Volkova, S., Shaffer, K., Jang, J.Y., Hodas, N.: Separating facts from fiction: Linguistic models to classify suspicious and trusted news posts on twitter. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). vol. 2, pp. 647–653 (2017)
21. Wu, Y., Agarwal, P.K., Li, C., Yang, J., Yu, C.: Toward computational fact-checking. Proceedings of the VLDB Endowment **7**(7), 589–600 (2014)
22. 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. p. 13. ACM (2012)
23. 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. pp. 1395–1405. International World Wide Web Conferences Steering Committee (2015)

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22734-0_25