Modeling Conversation Structure and Temporal Dynamics for Jointly Predicting Rumor Stance and Veracity

Penghui Wei, Nan Xu, Wenji Mao
[†]SKL-MCCS, Institute of Automation, Chinese Academy of Sciences (CASIA)
[‡]University of Chinese Academy of Sciences
{weipenghui2016,xunan2015,wenji.mao}@ia.ac.cn

Abstract

Automatically verifying rumorous information has become an important and challenging task in natural language processing and social media analytics. Previous studies reveal that people's stances towards rumorous messages can provide indicative clues for identifying the veracity of rumors, and thus determining the stances of public reactions is a crucial preceding step for rumor veracity prediction. In this paper, we propose a hierarchical multi-task learning framework for jointly predicting rumor stance and veracity on Twitter, which consists of two components. The bottom component of our framework classifies the stances of tweets in a conversation discussing a rumor via modeling the structural property based on a novel graph convolutional network. The top component predicts the rumor veracity by exploiting the temporal dynamics of stance evolution. Experimental results on two benchmark datasets show that our method outperforms previous methods in both rumor stance classification and veracity prediction.

1 Introduction

Social media websites have become the main platform for users to browse information and share opinions, facilitating news dissemination greatly. However, the characteristics of social media also accelerate the rapid spread and dissemination of unverified information, i.e., rumors (Shu et al., 2017). The definition of *rumor* is "*items of information that are unverified at the time of posting*" (Zubiaga et al., 2018). Ubiquitous false rumors bring about harmful effects, which has seriously affected public and individual lives, and caused panic in society (Zhou and Zafarani, 2018; Sharma et al., 2019). Because online content is massive and debunking rumors manually is time-consuming, there is a great need for automatic methods to identify false rumors (Oshikawa et al., 2018).

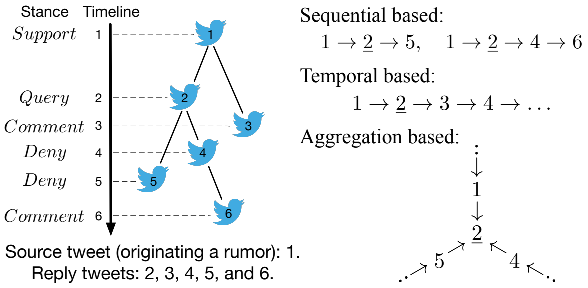


Figure 1: A conversation thread discussing the rumorous tweet "1". Three different perspectives for learning the stance feature of the reply tweet "2" are illustrated.

Previous studies have observed that public stances towards rumorous messages are crucial signals to detect trending rumors (Qazvinian et al., 2011; Zhao et al., 2015) and indicate the veracity of them (Mendoza et al., 2010; Procter et al., 2013; Liu et al., 2015; Jin et al., 2016; Glenski et al., 2018). Therefore, stance classification towards rumors is viewed as an important preceding step of rumor veracity prediction, especially in the context of Twitter conversations (Zubiaga et al., 2016a).

The state-of-the-art methods for rumor stance classification are proposed to model the sequential property (Kochkina et al., 2017) or the temporal property (Veyseh et al., 2017) of a Twitter conversation thread. In this paper, we propose a new perspective based on structural property: learning tweet representations through aggregating information from their neighboring tweets. Intuitively, a tweet's nearer neighbors in its conversation thread are more informative than farther neighbors because the replying relationships of them are closer, and their stance expressions can help classify the stance of the center tweet (e.g., in Figure 1, tweets "1", "4" and "5" are the one-hop neighbors of the tweet "2", and their influences on predicting the stance of "2" are larger than that of the two-hop neighbor "3"). To achieve this, we represent both

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 4787–4798,
Hong Kong, China, November 3–7, 2019. ©2019 Association for Computational Linguistics

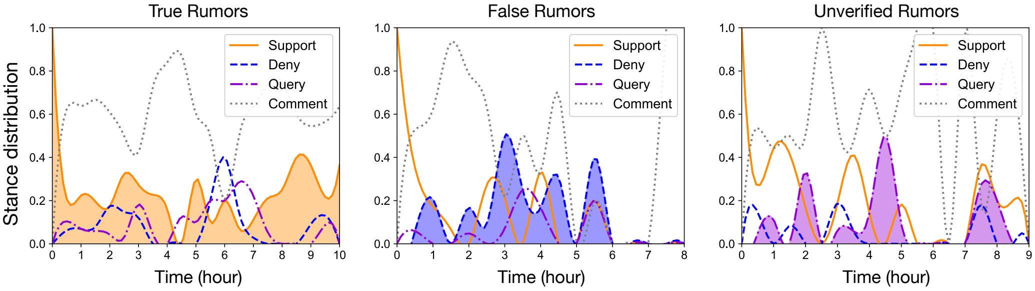


Figure 2: Stance distributions of tweets discussing $true$ rumors, $false$ rumors, and $unverified$ rumors, respectively (Better viewed in color). The horizontal axis is the spreading time of the first rumor. It is visualized based on SemEval-2017 task 8 dataset (Derczynski et al., 2017). All tweets are relevant to the event "Ottawa Shooting".

tweet contents and conversation structures into a latent space using a graph convolutional network (GCN) (Kipf and Welling, 2017), aiming to learn stance feature for each tweet by aggregating its neighbors' features. Compared with the sequential and temporal based methods, our aggregation based method leverages the intrinsic structural property in conversations to learn tweet representations.

After determining the stances of people's reactions, another challenge is how we can utilize public stances to predict rumor veracity accurately. We observe that the temporal dynamics of public stances can indicate rumor veracity. Figure 2 illustrates the stance distributions of tweets discussing $true$ rumors, $false$ rumors, and $unverified$ rumors, respectively. As we can see, $supporting$ stance dominates the inception phase of spreading. However, as time goes by, the proportion of $denying$ tweets towards $false$ rumors increases quite significantly. Meanwhile, the proportion of $querying$ tweets towards $unverified$ rumors also shows an upward trend. Based on this observation, we propose to model the temporal dynamics of stance evolution with a recurrent neural network (RNN), capturing the crucial signals containing in stance features for effective veracity prediction.

Further, most existing methods tackle stance classification and veracity prediction separately, which is suboptimal and limits the generalization of models. As shown previously, they are two closely related tasks in which stance classification can provide indicative clues to facilitate veracity prediction. Thus, these two tasks can be jointly learned to make better use of their interrelation.

Based on the above considerations, in this paper, we propose a hierarchical multi-task learning framework for jointly predicting rumor stance and veracity, which achieves deep integration between the preceding task (stance classification) and the subsequent task (veracity prediction). The bottom component of our framework classifies the stances of tweets in a conversation discussing a rumor via aggregation-based structure modeling, and we design a novel graph convolution operation customized for conversation structures. The top component predicts rumor veracity by exploiting the temporal dynamics of stance evolution, taking both content features and stance features learned by the bottom component into account. Two components are jointly trained to utilize the interrelation between the two tasks for learning more powerful feature representations.

The contributions of this work are as follows.

- We propose a hierarchical framework to tackle rumor stance classification and veracity prediction jointly, exploiting both structural characteristic and temporal dynamics in rumor spreading process.
- We design a novel graph convolution operation customized to encode conversation structures for learning stance features. To our knowledge, we are the first to employ graph convolution for modeling the structural property of Twitter conversations.
- Experimental results on two benchmark datasets verify that our hierarchical framework performs better than existing methods in both rumor stance classification and veracity prediction.

2 Related Work

Rumor Stance Classification Stance analysis has been widely studied in online debate forums (Somasundaran and Wiebe, 2009; Hasan and Ng, 2013), and recently has attracted increasing attention in different contexts (Mohammad et al., 2016; Augenstein et al., 2016; Ferreira and Vlachos, 2016; Mohtarami et al., 2018). After the pioneering studies on stance classification towards rumors in social media (Mendoza et al., 2010;

Qazvinian et al., 2011; Procter et al., 2013), linguistic feature (Hamidian and Diab, 2016; Zeng et al., 2016) and point process based methods (Lukasik et al., 2016, 2019) have been developed.

Recent work has focused on Twitter conversations discussing rumors. Zubiaga et al. (2016a) proposed to capture the sequential property of conversations with linear-chain CRF, and also used a tree-structured CRF to consider the conversation structure as a whole. Aker et al. (2017) developed a novel feature set that scores the level of users' confidence. Pamungkas et al. (2018) designed affective and dialogue-act features to cover various facets of affect. Giasemidis et al. (2018) proposed a semi-supervised method that propagates the stance labels on similarity graph. Beyond feature-based methods, Kochkina et al. (2017) utilized an LSTM to model the sequential branches in a conversation, and their system ranked the first in SemEval-2017 task 8. Veyseh et al. (2017) adopted attention to model the temporal property of a conversation and achieved the state-of-the-art performance.

Rumor Veracity Prediction Previous studies have proposed methods based on various features such as linguistics, time series and propagation structures (Castillo et al., 2011; Kwon et al., 2013; Wu et al., 2015; Ma et al., 2017). Neural networks show the effectiveness of modeling time series (Ma et al., 2016; Ruchansky et al., 2017) and propagation paths (Liu and Wu, 2018). Ma et al. (2018b)'s model adopted recursive neural networks to incorporate structure information into tweet representations and outperformed previous methods.

Some studies utilized stance labels as the input feature of veracity classifiers to improve the performance (Liu et al., 2015; Enayet and El-Beltagy, 2017). Dungs et al. (2018) proposed to recognize the temporal patterns of true and false rumors' stances by two hidden Markov models (HMMs). Unlike their solution, our method learns discriminative features of stance evolution with an RNN. Moreover, our method jointly predicts stance and veracity by exploiting both structural and temporal characteristics, whereas HMMs need stance labels as the input sequence of observations.

Joint Predictions of Rumor Stance and Veracity Several work has addressed the problem of jointly predicting rumor stance and veracity. These studies adopted multi-task learning to jointly train two tasks (Ma et al., 2018a; Kochkina et al., 2018; Poddar et al., 2018) and learned shared representations with parameter-sharing. Compared with such solutions based on "parallel" architectures, our method is deployed in a hierarchical fashion that encodes conversation structures to learn more powerful stance features by the bottom component, and models stance evolution by the top component, achieving deep integration between the two tasks' feature learning.

3 Problem Definition

Consider a Twitter conversation thread $\mathcal{C}$ which consists of a source tweet t_1 (originating a rumor) and a number of reply tweets $\{t_2, t_3, \ldots, t_{|\mathcal{C}|}\}$ that respond t_1 directly or indirectly, and each tweet t_i ($i \in [1, |\mathcal{C}|]$) expresses its stance towards the rumor. The thread $\mathcal{C}$ is a tree structure, in which the source tweet t_1 is the root node, and the replying relationships among tweets form the edges.[1]

This paper focuses on two tasks. The first task is rumor stance classification, aiming to determine the stance of each tweet in $\mathcal{C}$, which belongs to $\{supporting, denying, querying, commenting\}$. The second task is rumor veracity prediction, with the aim of identifying the veracity of the rumor, belonging to $\{true, false, unverified\}$.

4 Proposed Method

We propose a Hierarchical multi-task learning framework for jointly Predicting rumor Stance and Veracity (named *Hierarchical-PSV*). Figure 3 illustrates its overall architecture that is composed of two components. The bottom component is to classify the stances of tweets in a conversation thread, which learns stance features via encoding conversation structure using a customized graph convolutional network (named *Conversational-GCN*). The top component is to predict the rumor's veracity, which takes the learned features from the bottom component into account and models the temporal dynamics of stance evolution with a recurrent neural network (named *Stance-Aware RNN*).

4.1 Conversational-GCN: Aggregation-based Structure Modeling for Stance Prediction

Now we detail *Conversational-GCN*, the bottom component of our framework. We first adopt a bidirectional GRU (BGRU) (Cho et al., 2014) layer to learn the content feature for each tweet in the thread $\mathcal{C}$. For a tweet t_i ($i \in [1, |\mathcal{C}|]$), we run the

[1] We consider each replying relationship between two tweets in a conversation as an undirected edge in the tree.

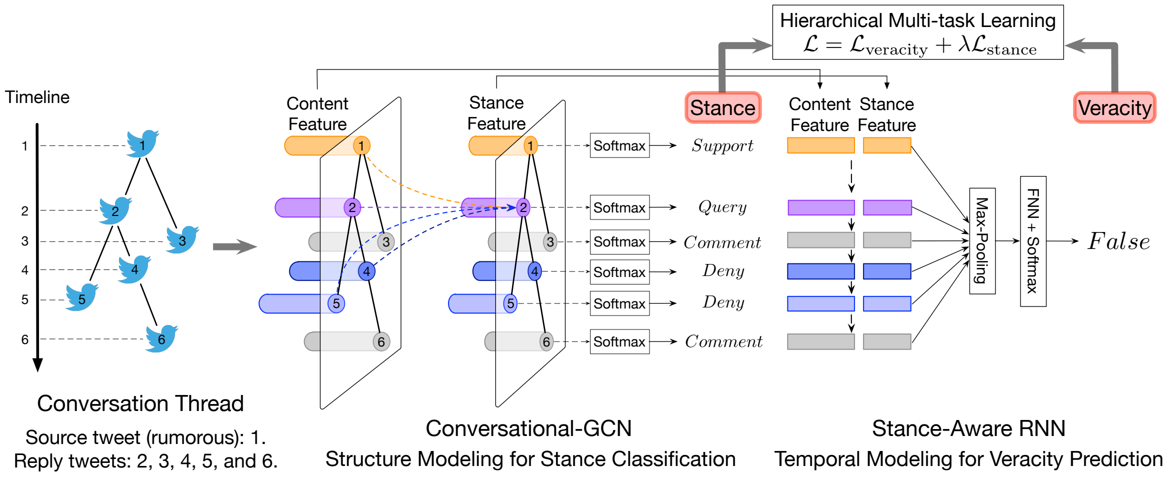

Hierarchical Multi-task Learning
$\mathcal{L} = \mathcal{L}_{\text{veracity}} + \lambda \mathcal{L}_{\text{stance}}$
Timeline
1
2
3
4
5
6
Content Feature
Stance Feature
Stance
Veracity
Content Feature
Stance Feature
Softmax
Support
Query
Comment
Deny
Deny
Comment
Max-Pooling
FNN + Softmax
False
Conversation Thread
Source tweet (rumorous): 1.
Reply tweets: 2, 3, 4, 5, and 6.
Conversational-GCN
Structure Modeling for Stance Classification
Stance-Aware RNN
Temporal Modeling for Veracity Prediction

Figure 3: Overall architecture of our proposed framework for joint predictions of rumor stance and veracity. In this illustration, the number of GCN layers is one. The information aggregation process for the tweet t_2 based on original graph convolution operation (Eq. (1)) is detailed.

BGRU over its word embedding sequence, and use the final step's hidden vector to represent the tweet. The content feature representation of t_i is denoted as $\boldsymbol{c}_i \in \mathbb{R}^d$, where d is the output size of the BGRU.

As we mentioned in Section 1, the stance expressions of a tweet t_i's nearer neighbors can provide more informative signals than farther neighbors for learning t_i's stance feature. Based on the above intuition, we model the structural property of the conversation thread $\mathcal{C}$ to learn stance feature representation for each tweet in $\mathcal{C}$. To this end, we encode structural contexts to improve tweet representations by aggregating information from neighboring tweets with a graph convolutional network (GCN) (Kipf and Welling, 2017).

Formally, the conversation $\mathcal{C}$'s structure can be represented by a graph $\mathcal{C}_G = \langle \mathcal{T}, \mathcal{E} \rangle$, where $\mathcal{T} = \{t_i\}_{i=1}^{|\mathcal{C}|}$ denotes the node set (i.e., tweets in the conversation), and $\mathcal{E}$ denotes the edge set composed of all replying relationships among the tweets. We transform the edge set $\mathcal{E}$ to an adjacency matrix $\mathbf{A} \in \mathbb{R}^{|\mathcal{C}| \times |\mathcal{C}|}$, where $\mathbf{A}_{ij} = \mathbf{A}_{ji} = 1$ if the tweet t_i directly replies the tweet t_j or $i = j$. In one GCN layer, the graph convolution operation for one tweet t_i on $\mathcal{C}_G$ is defined as:

$$\boldsymbol{h}_i^{\text{out}} = \tanh\left(\sum_{j \in \{j \mid \hat{\mathbf{A}}_{ij} \neq 0\}} \hat{\mathbf{A}}_{ij} \boldsymbol{W}^\top \boldsymbol{h}_j^{\text{in}} + \boldsymbol{b} \right), \tag{1}$$

where $\boldsymbol{h}_i^{\text{in}} \in \mathbb{R}^{d_{\text{in}}}$ and $\boldsymbol{h}_i^{\text{out}} \in \mathbb{R}^{d_{\text{out}}}$ denote the input and output feature representations of the tweet t_i respectively. The convolution filter $\boldsymbol{W} \in \mathbb{R}^{d_{\text{in}} \times d_{\text{out}}}$ and the bias $\boldsymbol{b} \in \mathbb{R}^{d_{\text{out}}}$ are shared over all tweets in a conversation. We apply symmetric normalized transformation $\hat{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}$ to avoid the scale changing of feature representations, where $\mathbf{D}$ is the degree matrix of $\mathbf{A}$, and $\{j \mid \hat{\mathbf{A}}_{ij} \neq 0\}$ contains t_i's one-hop neighbors and t_i itself.

In this original graph convolution operation, given a tweet t_i, the receptive field for t_i contains its one-hop neighbors and t_i itself, and the aggregation level of two tweets t_i and t_j is dependent on $\hat{\mathbf{A}}_{ij}$. In the context of encoding conversation structures, we observe that such operation can be further improved for two issues. First, a tree-structured conversation may be very deep, which means that the receptive field of a GCN layer is restricted in our case. Although we can stack multiple GCN layers to expand the receptive field, it is still difficult to handle conversations with deep structures and increases the number of parameters. Second, the normalized matrix $\hat{\mathbf{A}}$ partly weakens the importance of the tweet t_i itself. To address these issues, we design a novel graph convolution operation which is customized to encode conversation structures. Formally, it is implemented by modifying the matrix $\hat{\mathbf{A}}$ in Eq. (1):

$$\hat{\mathbf{A}} \leftarrow \hat{\mathbf{A}}\hat{\mathbf{A}} + \mathbf{I}\,, \tag{2}$$

where the multiplication operation expands the receptive field of a GCN layer, and adding an identity matrix elevates the importance of t_i itself.

After defining the above graph convolution operation, we adopt an L-layer GCN to model conversation structures. The l^{th} GCN layer ($l \in [1, L]$) computed over the entire conversation structure can be written as an efficient matrix operation:

$$\boldsymbol{H}^{(l)} = \tanh(\hat{\mathbf{A}}\boldsymbol{H}^{(l-1)}\boldsymbol{W}^{(l)} + \boldsymbol{b}^{(l)}) , \quad (3)$$

where $\boldsymbol{H}^{(l-1)} \in \mathbb{R}^{|\mathcal{C}| \times d_{l-1}}$ and $\boldsymbol{H}^{(l)} \in \mathbb{R}^{|\mathcal{C}| \times d_l}$ denote the input and output features of all tweets in the conversation $\mathcal{C}$ respectively.

Specifically, the first GCN layer takes the content features of all tweets as input, i.e., $\boldsymbol{H}^{(0)} = (\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_{|\mathcal{C}|})^\top \in \mathbb{R}^{|\mathcal{C}| \times d}$. The output of the last GCN layer represents the stance features of all tweets in the conversation, i.e., $\boldsymbol{H}^{(L)} = (\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_{|\mathcal{C}|})^\top \in \mathbb{R}^{|\mathcal{C}| \times 4}$, where $\boldsymbol{s}_i$ is the unnormalized stance distribution of the tweet t_i.

For each tweet t_i in the conversation $\mathcal{C}$, we apply softmax to obtain its predicted stance distribution:

$$\hat{\boldsymbol{s}}_i = \mathrm{softmax}(\boldsymbol{s}_i) \in \mathbb{R}^4 , \quad i \in [1, |\mathcal{C}|] . \quad (4)$$

The ground-truth labels of stance classification supervise the learning process of *Conversational-GCN*. The loss function of $\mathcal{C}$ for stance classification is computed by cross-entropy criterion:

$$\mathcal{L}_{\text{stance}} = \frac{1}{|\mathcal{C}|} \sum_{i=1}^{|\mathcal{C}|} \left(-s_i^\top \log \hat{\boldsymbol{s}}_i \right) , \quad (5)$$

where s_i is a one-hot vector that denotes the stance label of the tweet t_i. For batch-wise training, the objective function for a batch is the averaged cross-entropy loss of all tweets in these conversations.

In previous studies, GCNs are used to encode dependency trees (Marcheggiani and Titov, 2017; Zhang et al., 2018) and cross-document relations (Yasunaga et al., 2017; De Cao et al., 2019) for downstream tasks. Our work is the first to leverage GCNs for encoding conversation structures.

4.2 Stance-Aware RNN: Temporal Dynamics Modeling for Veracity Prediction

The top component, *Stance-Aware RNN*, aims to capture the temporal dynamics of stance evolution in a conversation discussing a rumor. It integrates both content features and stance features learned from the bottom *Conversational-GCN* to facilitate the veracity prediction of the rumor.

Specifically, given a conversation thread $\mathcal{C} = \{t_1, t_2, \ldots, t_{|\mathcal{C}|}\}$ (where the tweets t_* are ordered chronologically), we combine the content feature and the stance feature for each tweet, and adopt a GRU layer to model the temporal evolution:

$$\boldsymbol{v}_i = \mathrm{GRU}([\boldsymbol{c}_i; \boldsymbol{s}_i], \boldsymbol{v}_{i-1}) , \quad i \in [1, |\mathcal{C}|] , \quad (6)$$

where $[\cdot;\cdot]$ denotes vector concatenation, and $(\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{|\mathcal{C}|})$ is the output sequence that represents the temporal feature. We then transform the sequence to a vector $\boldsymbol{v}$ by a max-pooling function that captures the global information of stance evolution, and feed it into a one-layer feed-forward neural network (FNN) with softmax normalization to produce the predicted veracity distribution $\hat{\boldsymbol{v}}$:

$$\begin{aligned} \boldsymbol{v} &= \text{max-pooling}(\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{|\mathcal{C}|}) , \\ \hat{\boldsymbol{v}} &= \mathrm{softmax}(\mathrm{FNN}(\boldsymbol{v})) . \end{aligned} \quad (7)$$

The loss function of $\mathcal{C}$ for veracity prediction is also computed by cross-entropy criterion:

$$\mathcal{L}_{\text{veracity}} = -v^\top \log \hat{\boldsymbol{v}} , \quad (8)$$

where v denotes the veracity label of $\mathcal{C}$.

4.3 Jointly Learning Two Tasks

To leverage the interrelation between the preceding task (stance classification) and the subsequent task (veracity prediction), we jointly train two components in our framework. Specifically, we add two tasks' loss functions to obtain a joint loss function $\mathcal{L}$ (with a trade-off parameter λ), and optimize $\mathcal{L}$ to train our framework:

$$\mathcal{L} = \mathcal{L}_{\text{veracity}} + \lambda \mathcal{L}_{\text{stance}} . \quad (9)$$

In our *Hierarchical-PSV*, the bottom component *Conversational-GCN* learns content and stance features, and the top component *Stance-Aware RNN* takes the learned features as input to further exploit temporal evolution for predicting rumor veracity. Our multi-task framework achieves deep integration of the feature representation learning process for the two closely related tasks.

5 Experiments

In this section, we first evaluate the performance of *Conversational-GCN* on rumor stance classification and evaluate *Hierarchical-PSV* on veracity prediction (Section 5.3). We then give a detailed analysis of our proposed method (Section 5.4).

5.1 Data & Evaluation Metric

To evaluate our proposed method, we conduct experiments on two benchmark datasets.

The first is **SemEval-2017 task 8** (Derczynski et al., 2017) dataset. It includes 325 rumorous conversation threads, and has been split into training, development and test sets. These threads cover ten

Dataset	# Thread	Depth	# Tweet	Stance Labels				Veracity Labels		
				# support	*# deny*	*# query*	*# comment*	*# true*	*# false*	*# unverified*
SemEval	325	3.5	5,568	1,004	415	464	3,685	145	74	106
PHEME	2,402	2.8	105,354	–				1,067	638	697

Table 1: Statistics of two datasets. The column "Depth" denotes the average depth of all conversation threads.

Method	Evaluation Metric					
	Macro-F_1	F_S	F_D	F_Q	F_C	Acc.
Affective Feature + SVM (Pamungkas et al., 2018)	0.470	**0.410**	0.000	0.580	**0.880**	0.795
BranchLSTM (Kochkina et al., 2017)	0.434	0.403	0.000	0.462	0.873	0.784
TemporalAttention (Veyseh et al., 2017)	0.482	–	–	–	–	**0.820**
Conversational-GCN (Ours, $L = 2$)	**0.499**	0.311	**0.194**	**0.646**	0.847	0.751

Table 2: Results of rumor stance classification. F_S, F_D, F_Q and F_C denote the F_1 scores of $supporting$, $denying$, $querying$ and $commenting$ classes respectively. "–" indicates that the original paper does not report the metric.

events, and two events of that only appear in the test set. This dataset is used to evaluate both stance classification and veracity prediction tasks.

The second is **PHEME** dataset (Zubiaga et al., 2016b). It provides 2,402 conversations covering nine events. Following previous work, we conduct leave-one-event-out cross-validation: in each fold, one event's conversations are used for testing, and all the rest events are used for training. The evaluation metric on this dataset is computed after integrating the outputs of all nine folds. Note that only a subset of this dataset has stance labels, and all conversations in this subset are already contained in SemEval-2017 task 8 dataset. Thus, PHEME dataset is used to evaluate veracity prediction task.

Table 1 shows the statistics of two datasets. Because of the class-imbalanced problem, we use macro-averaged F_1 as the evaluation metric for two tasks. We also report accuracy for reference.

5.2 Implementation Details

In all experiments, the number of GCN layers is set to $L = 2$. We list the implementation details in Appendix A.

5.3 Experimental Results

5.3.1 Results: Rumor Stance Classification

Baselines We compare our *Conversational-GCN* with the following methods in the literature:

• *Affective Feature + SVM* (Pamungkas et al., 2018) extracts affective and dialogue-act features for individual tweets, and then trains an SVM for classifying stances.

• *BranchLSTM* (Kochkina et al., 2017) is the winner of SemEval-2017 shared task 8 subtask A. It adopts an LSTM to model the sequential branches in a conversation thread. Before feeding branches into the LSTM, some additional hand-crafted features are used to enrich the tweet representations.

• *TemporalAttention* (Veyseh et al., 2017) is the state-of-the-art method. It uses a tweet's "neighbors in the conversation timeline" as the context, and utilizes attention to model such temporal sequence for learning the weight of each neighbor. Extra hand-crafted features are also used.

Performance Comparison Table 2 shows the results of different methods for rumor stance classification. Clearly, the macro-averaged F_1 of Conversational-GCN is better than all baselines.

Especially, our method shows the effectiveness of determining $denying$ stance, while other methods can not give any correct prediction for $denying$ class (the F_D scores of them are equal to zero). Further, Conversational-GCN also achieves higher F_1 score for $querying$ stance (F_Q). Identifying $denying$ and $querying$ stances correctly is crucial for veracity prediction because they play the role of indicators for $false$ and $unverified$ rumors respectively (see Figure 2). Meanwhile, the class-imbalanced problem of data makes this a challenge. Conversational-GCN effectively encodes structural context for each tweet via aggregating information from its neighbors, learning powerful stance features without feature engineering. It is also more computationally efficient than sequential and temporal based methods. The information aggregations for all tweets in a conversation are worked in parallel and thus the running time is not sensitive to conversation's depth.

Setting	Method	SemEval dataset		PHEME dataset	
		Macro-F_1	Acc.	Macro-F_1	Acc.
Single-task	TD-RvNN (Ma et al., 2018b)	0.509	0.536	0.264	0.341
	Hierarchical GCN-RNN (Ours)	0.540	0.536	0.317	0.356
Multi-task	BranchLSTM+NileTMRG (Kochkina et al., 2018)	0.539	0.570	0.297	0.360
	MTL2 (Veracity+Stance) (Kochkina et al., 2018)	0.558	0.571	0.318	0.357
	Hierarchical-PSV (Ours, $\lambda = 1$)	**0.588**	**0.643**	**0.333**	**0.361**

Table 3: Results of veracity prediction. Single-task setting means that stance labels cannot be used to train models.

5.3.2 Results: Rumor Veracity Prediction

To evaluate our framework *Hierarchical-PSV*, we consider two groups of baselines: single-task and multi-task baselines.

Single-task Baselines In single-task setting, stance labels are not available. Only veracity labels can be used to supervise the training process.

• *TD-RvNN* (Ma et al., 2018b) models the top-down tree structure using a recursive neural network for veracity classification.

• *Hierarchical GCN-RNN* is the single-task variant of our framework: we optimize $\mathcal{L}_{\text{veracity}}$ (i.e., $\lambda = 0$ in Eq. (9)) during training. Thus, the bottom Conversational-GCN only has indirect supervision (veracity labels) to learn stance features.

Multi-task Baselines In multi-task setting, both stance labels and veracity labels are available for training.

• *BranchLSTM+NileTMRG* (Kochkina et al., 2018) is a pipeline method, combining the winner systems of two subtasks in SemEval-2017 shared task 8. It first trains a BranchLSTM for stance classification, and then uses the predicted stance labels as extra features to train an SVM for veracity prediction (Enayet and El-Beltagy, 2017).

• *MTL2 (Veracity+Stance)* (Kochkina et al., 2018) is a multi-task learning method that adopts BranchLSTM as the shared block across tasks. Then, each task has a task-specific output layer, and two tasks are jointly learned.[2]

Performance Comparison Table 3 shows the comparisons of different methods. By comparing single-task methods, Hierarchical GCN-RNN performs better than TD-RvNN, which indicates that our hierarchical framework can effectively model conversation structures to learn high-quality tweet representations. The recursive operation in TD-RvNN is performed in a fixed direction and runs over all tweets, thus may not obtain enough useful information. Moreover, the training speed of Hierarchical GCN-RNN is significantly faster than TD-RvNN: in the condition of batch-wise optimization for training one step over a batch containing 32 conversations, our method takes only 0.18 seconds, while TD-RvNN takes 5.02 seconds.

Comparisons among multi-task methods show that two joint methods outperform the pipeline method (BranchLSTM+NileTMRG), indicating that jointly learning two tasks can improve the generalization through leveraging the interrelation between them. Further, compared with MTL2 which uses a "parallel" architecture to make predictions for two tasks, our Hierarchical-PSV performs better than MTL2. The hierarchical architecture is more effective to tackle the joint predictions of rumor stance and veracity, because it not only possesses the advantage of parameter-sharing but also offers deep integration of the feature representation learning process for the two tasks. Compared with Hierarchical GCN-RNN that does not use the supervision from stance classification task, Hierarchical-PSV provides a performance boost, which demonstrates that our framework benefits from the joint learning scheme.

5.4 Further Analysis and Discussions

We conduct additional experiments to further demonstrate the effectiveness of our model.

5.4.1 Effect of Customized Graph Convolution

To show the effect of our customized graph convolution operation (Eq. (2)) for modeling conversation structures, we further compare it with the original graph convolution (Eq. (1), named Original-GCN) on stance classification task.

Specifically, we cluster tweets in the test set according to their depths in the conversation threads (e.g., the cluster "depth = 0" consists of all source

[2] Kochkina et al. (2018) also proposed *MTL3* that jointly trains three tasks (plus rumor detection (Zubiaga et al., 2017)). In our framework, we do not utilize data and labels from rumor detection task, and thus we choose *MTL2 (Veracity+Stance)* as the state-of-the-art method for comparison.

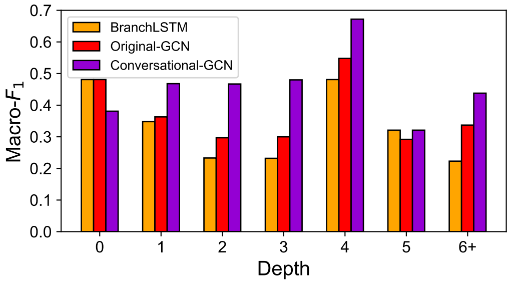


Figure 4: Stance classification results w.r.t. different depths (see Appendix B for exact numerical numbers).

Method	Macro-F_1	Acc.
Hierarchical-PSV (full model)	**0.333**	**0.361**
– stance features	0.299	0.338
– GRU, + CNN	0.312	0.328
– GRU	0.288	0.326

Table 4: Ablation tests of stance features and temporal modeling for veracity prediction on PHEME dataset.

tweets in the test set). For BranchLSTM, Original-GCN and Conversational-GCN, we report their macro-averaged F_1 on each cluster in Figure 4.

We observe that our Conversational-GCN outperforms Original-GCN and BranchLSTM significantly in most levels of depth. BranchLSTM may prefer to “shallow” tweets in a conversation because they often occur in multiple branches (e.g., in Figure 1, the tweet “2” occurs in two branches and thus it will be modeled twice). The results indicate that Conversational-GCN has advantage to identify stances of “deep” tweets in conversations.

5.4.2 Ablation Tests

Effect of Stance Features To understand the importance of stance features for veracity prediction, we conduct an ablation study: we only input the content features of all tweets in a conversation to the top component RNN. It means that the RNN only models the temporal variation of tweet contents during spreading, but does not consider their stances and is not “stance-aware”. Table 4 shows that “– stance features” performs poorly, and thus the temporal modeling process benefits from the indicative signals provided by stance features. Hence, combining the low-level content features and the high-level stance features is crucial to improve rumor veracity prediction.

Effect of Temporal Evolution Modeling We modify the Stance-Aware RNN by two ways: (i) we replace the GRU layer by a CNN that only captures local temporal information; (ii) we remove the GRU layer. Results in Table 4 verify that replacing or removing the GRU block hurts the performance, and thus modeling the stance evolution of public reactions towards a rumorous message is indeed necessary for effective veracity prediction.

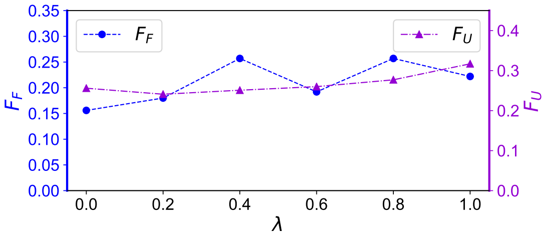
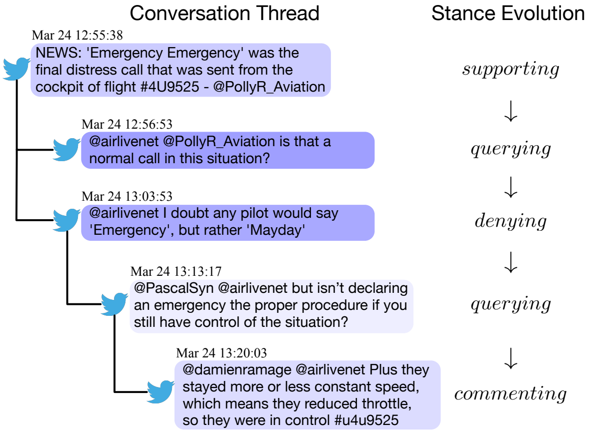


Figure 5: Veracity prediction results v.s. various values of λ on PHEME dataset. F_{F} and F_{U} denote the F_1 scores of $false$ and $unverified$ classes respectively.



Mar 24 12:55:38
NEWS: 'Emergency Emergency' was the final distress call that was sent from the cockpit of flight #4U9525 - @PollyR_Aviation

Mar 24 12:56:53
@airlivenet @PollyR_Aviation is that a normal call in this situation?

Mar 24 13:03:53
@airlivenet I doubt any pilot would say 'Emergency', but rather 'Mayday'

Mar 24 13:13:17
@PascalSyn @airlivenet but isn’t declaring an emergency the proper procedure if you still have control of the situation?

Mar 24 13:20:03
@damienramage @airlivenet Plus they stayed more or less constant speed, which means they reduced throttle, so they were in control #u4u9525

Figure 6: Case study: a $false$ rumor. Each tweet is colored by the number of dimensions it contributes to $\boldsymbol{v}$ in the max-pooling operation (Eq. (7)). We show important tweets in the conversation and truncate others.

5.4.3 Interrelation of Stance and Veracity

We vary the value of λ in the joint loss $\mathcal{L}$ and train models with various λ to show the interrelation between stance and veracity in Figure 5. As λ increases from 0.0 to 1.0, the performance of identifying $false$ and $unverified$ rumors generally gains. Therefore, when the supervision signal of stance classification becomes strong, the learned stance features can produce more accurate clues for predicting rumor veracity.

5.5 Case Study

Figure 6 illustrates a $false$ rumor identified by our model. We can observe that the stances of reply tweets present a typical temporal pattern “$supporting \rightarrow querying \rightarrow denying$”. Our model captures such stance evolution with RNN and predicts its veracity correctly. Further, the visualization of tweets shows that the max-pooling operation catches informative tweets in the conversation. Hence, our framework can notice salience indicators of rumor veracity in the spreading process and combine them to give correct prediction.

6 Conclusion

We propose a hierarchical multi-task learning framework for jointly predicting rumor stance and veracity on Twitter. We design a new graph convolution operation, Conversational-GCN, to encode conversation structures for classifying stance, and then the top Stance-Aware RNN combines the learned features to model the temporal dynamics of stance evolution for veracity prediction. Experimental results verify that Conversational-GCN can handle deep conversation structures effectively, and our hierarchical framework performs much better than existing methods. In future work, we shall explore to incorporate external context (Derczynski et al., 2017; Popat et al., 2018), and extend our model to multi-lingual scenarios (Wen et al., 2018). Moreover, we shall investigate the diffusion process of rumors from social science perspective (Vosoughi et al., 2018), draw deeper insights from there and try to incorporate them into the model design.

Acknowledgments

This work was supported in part by the National Key R&D Program of China under Grant #2016QY02D0305, NSFC Grants #71621002, #71472175, #71974187 and #71602184, and Ministry of Health of China under Grant #2017ZX10303401-002. We thank all the anonymous reviewers for their valuable comments. We also thank Qianqian Dong for her kind assistance.

References

Ahmet Aker, Leon Derczynski, and Kalina Bontcheva. 2017. Simple open stance classification for rumour analysis. In *Proceedings of RANLP*, pages 31–39.

Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. 2016. Stance detection with bidirectional conditional encoding. In *Proceedings of EMNLP*, pages 876–885.

Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. 2011. Information credibility on Twitter. In *Proceedings of WWW*, pages 675–684.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of EMNLP*, pages 1724–1734.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2019. Question answering by reasoning across documents with graph convolutional networks. In *Proceedings of NAACL-HLT*, pages 2306–2317.

Leon Derczynski, Kalina Bontcheva, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Arkaitz Zubiaga. 2017. SemEval-2017 task 8: RumourEval: Determining rumour veracity and support for rumours. In *Proceedings of SemEval*, pages 69–76.

Sebastian Dungs, Ahmet Aker, Norbert Fuhr, and Kalina Bontcheva. 2018. Can rumour stance alone predict veracity? In *Proceedings of COLING*, pages 3360–3370.

Omar Enayet and Samhaa R El-Beltagy. 2017. NileTMRG at SemEval-2017 task 8: Determining rumour and veracity support for rumours on Twitter. In *Proceedings of SemEval*, pages 470–474.

William Ferreira and Andreas Vlachos. 2016. Emergent: A novel data-set for stance classification. In *Proceedings of NAACL-HLT*, pages 1163–1168.

Georgios Giasemidis, Nikolaos Kaplis, Ioannis Agrafiotis, and Jason Nurse. 2018. A semi-supervised approach to message stance classification. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*.

Maria Glenski, Tim Weninger, and Svitlana Volkova. 2018. Identifying and understanding user reactions to deceptive and trusted social news sources. In *Proceedings of ACL*, pages 176–181.

Sardar Hamidian and Mona Diab. 2016. Rumor identification and belief investigation on Twitter. In *Proceedings of NAACL-HLT*, pages 3–8.

Kazi Saidul Hasan and Vincent Ng. 2013. Extra-linguistic constraints on stance recognition in ideological debates. In *Proceedings of ACL*, pages 816–821.

Zhiwei Jin, Juan Cao, Yongdong Zhang, and Jiebo Luo. 2016. News verification by exploiting conflicting social viewpoints in microblogs. In *Proceedings of AAAI*, pages 2972–2978.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of EMNLP*, pages 1746–1751.

Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of ICLR*.

Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *Proceedings of ICLR*.

Elena Kochkina, Maria Liakata, and Isabelle Augenstein. 2017. Turing at SemEval-2017 task 8: Sequential approach to rumour stance classification with branch-LSTM. In *Proceedings of SemEval*, pages 475–480.

Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. 2018. All-in-one: Multi-task learning for rumour verification. In *Proceedings of COLING*, pages 3402–3413.

Sejeong Kwon, Meeyoung Cha, Kyomin Jung, Wei Chen, and Yajun Wang. 2013. Prominent features of rumor propagation in online social media. In *Proceedings of ICDM*, pages 1103–1108.

Xiaomo Liu, Armineh Nourbakhsh, Quanzhi Li, Rui Fang, and Sameena Shah. 2015. Real-time rumor debunking on Twitter. In *Proceedings of CIKM*, pages 1867–1870.

Yang Liu and Yi-fang Brook Wu. 2018. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Proceedings of AAAI*, pages 354–361.

Michal Lukasik, Kalina Bontcheva, Trevor Cohn, Arkaitz Zubiaga, Maria Liakata, and Rob Procter. 2019. Gaussian processes for rumour stance classification in social media. *ACM Transactions on Information Systems (TOIS)*, 37(2):20.

Michal Lukasik, PK Srijith, Duy Vu, Kalina Bontcheva, Arkaitz Zubiaga, and Trevor Cohn. 2016. Hawkes processes for continuous time sequence classification: An application to rumour stance classification in Twitter. In *Proceedings of ACL*, pages 393–398.

Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. 2016. Detecting rumors from microblogs with recurrent neural networks. In *Proceedings of IJCAI*, pages 3818–3824.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2017. Detect rumors in microblog posts using propagation structure via kernel learning. In *Proceedings of ACL*, pages 708–717.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2018a. Detect rumor and stance jointly by neural multi-task learning. In *Proceedings of WWW Companion*, pages 585–593.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2018b. Rumor detection on Twitter with tree-structured recursive neural networks. In *Proceedings of ACL*, pages 1980–1989.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. In *Proceedings of EMNLP*, pages 1506–1515.

Marcelo Mendoza, Barbara Poblete, and Carlos Castillo. 2010. Twitter under crisis: Can we trust what we RT? In *Proceedings of the 1st Workshop on Social Media Analytics (SOMA)*, pages 71–79.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*, pages 3111–3119.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. SemEval-2016 task 6: Detecting stance in tweets. In *Proceedings of SemEval*, pages 31–41.

Mitra Mohtarami, Ramy Baly, James Glass, Preslav Nakov, Lluís Màrquez, and Alessandro Moschitti. 2018. Automatic stance detection using end-to-end memory networks. In *Proceedings of NAACL-HLT*, pages 767–776.

Ray Oshikawa, Jing Qian, and William Yang Wang. 2018. A survey on natural language processing for fake news detection. *arXiv preprint arXiv:1811.00770*.

Endang Wahyu Pamungkas, Valerio Basile, and Viviana Patti. 2018. Stance classification for rumour analysis in Twitter: Exploiting affective information and conversation structure. In *Proceedings of the 2nd International Workshop on Rumours and Deception in Social Media (RDSM)*.

Lahari Poddar, Wynne Hsu, Mong Li Lee, and Shruti Subramaniyam. 2018. Predicting stances in Twitter conversations for detecting veracity of rumors: A neural approach. In *Proceedings of ICTAI*, pages 65–72.

Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. 2018. DeClarE: Debunking fake news and false claims using evidence-aware deep learning. In *Proceedings of EMNLP*, pages 22–32.

Rob Procter, Farida Vis, and Alex Voss. 2013. Reading the riots on Twitter: Methodological innovation for the analysis of big data. *International Journal of Social Research Methodology*, 16(3):197–214.

Vahed Qazvinian, Emily Rosengren, Dragomir R Radev, and Qiaozhu Mei. 2011. Rumor has it: Identifying misinformation in microblogs. In *Proceedings of EMNLP*, pages 1589–1599.

Natali Ruchansky, Sungyong Seo, and Yan Liu. 2017. CSI: A hybrid deep model for fake news detection. In *Proceedings of CIKM*, pages 797–806.

Karishma Sharma, Feng Qian, He Jiang, Natali Ruchansky, Ming Zhang, and Yan Liu. 2019. Combating fake news: A survey on identification and mitigation techniques. *ACM Transactions on Intelligent Systems and Technology (TIST)*.

Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 19(1):22–36.

Swapna Somasundaran and Janyce Wiebe. 2009. Recognizing stances in online debates. In *Proceedings of ACL-IJCNLP*, pages 226–234.

Amir Pouran Ben Veyseh, Javid Ebrahimi, Dejing Dou, and Daniel Lowd. 2017. A temporal attentional model for rumor stance classification. In *Proceedings of CIKM*, pages 2335–2338.

Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *Science*, 359(6380):1146–1151.

Weiming Wen, Songwen Su, and Zhou Yu. 2018. Cross-lingual cross-platform rumor verification pivoting on multimedia content. In *Proceedings of EMNLP*, pages 3487–3496.

Ke Wu, Song Yang, and Kenny Q Zhu. 2015. False rumors detection on Sina weibo by propagation structures. In *Proceedings of ICDE*, pages 651–662.

Michihiro Yasunaga, Rui Zhang, Kshitijh Meelu, Ayush Pareek, Krishnan Srinivasan, and Dragomir Radev. 2017. Graph-based neural multi-document summarization. In *Proceedings of CoNLL*, pages 452–462.

Li Zeng, Kate Starbird, and Emma S Spiro. 2016. #unconfirmed: Classifying rumor stance in crisis-related social media messages. In *Proceedings of ICWSM*, pages 747–750.

Yuhao Zhang, Peng Qi, and Christopher D Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of EMNLP*, pages 2205–2215.

Zhe Zhao, Paul Resnick, and Qiaozhu Mei. 2015. Enquiring minds: Early detection of rumors in social media from enquiry posts. In *Proceedings of WWW*, pages 1395–1405.

Xinyi Zhou and Reza Zafarani. 2018. Fake news: A survey of research, detection methods, and opportunities. *arXiv preprint arXiv:1812.00315*.

Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. 2018. Detection and resolution of rumours in social media: A survey. *ACM Computing Surveys (CSUR)*, 51(2):32.

Arkaitz Zubiaga, Elena Kochkina, Maria Liakata, Rob Procter, and Michal Lukasik. 2016a. Stance classification in rumours as a sequential task exploiting the tree structure of social media conversations. In *Proceedings of COLING*, pages 2438–2448.

Arkaitz Zubiaga, Maria Liakata, and Rob Procter. 2017. Exploiting context for rumour detection in social media. In *Proceedings of SocInfo*, pages 109–123.

Arkaitz Zubiaga, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Peter Tolmie. 2016b. Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PloS one*, 11(3):e0150989.

A Implementation Details

A.1 Hyperparameters

SemEval-2017 task 8 dataset We pretrain 200-dimensional word embeddings by Skip-gram with negative sampling (Mikolov et al., 2013), and they are fixed during the training process. We set the dimension of content feature to 200. We use a two-layer GCN, and the output sizes of two layers are 200 and 4, respectively. The max-pooling function in Eq. (7) is to select the maximum value of each dimension, and thus the size of $\boldsymbol{v}$ is equal to that of $\boldsymbol{v}_i$. The trade-off parameter λ is set to 1. We add dropout with 0.5 ratio to all but the last GCN layers and the FNN layer (used for veracity prediction). We train our model with 0.001 learning rate and 32 batch size using Adam (Kingma and Ba, 2015). For this dataset, after tuning hyperparameters on the development set, we merge the training and the development sets and re-train our model on the merged set.

PHEME dataset We set the learning rate to 0.005 for accelerating the training process. Other configurations are same to that of SemEval dataset. Because only a subset of PHEME dataset contains stance labels, if a training conversation does not have stance labels, we will not compute its loss function of rumor stance classification task during the training process.

A.2 The CNN Layer in Ablation Study

In Section 5.4.2, to demonstrate the effectiveness of modeling the temporal dynamics of stance evolution, we replace the GRU layer by a CNN layer that only captures local temporal information. Specifically, this CNN layer consists of a 1D convolution layer and a max-pooling function (Kim, 2014). We use three different filter windows: 2, 3 and 4. Each filter window has 100 feature maps. The output vector of this CNN layer is then fed into an FNN layer with softmax function to obtain the predicted veracity distribution.

B Numerical Results of Figure 4

Table 5 shows the exact numerical numbers of the results in Figure 4.

Depth	**Method**		
	BranchLSTM	Original-GCN	Ours
0	**0.481**	**0.481**	0.381
1	0.348	0.363	**0.468**
2	0.233	0.297	**0.467**
3	0.232	0.300	**0.480**
4	0.481	0.548	**0.672**
5	**0.321**	0.292	**0.321**
6+	0.223	0.337	**0.438**

Table 5: Stance classification results w.r.t. different depths.