

“© 2023 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.”

# Rumor Detection with Hierarchical Representation on Bipartite Adhoc Event Trees

Qi Zhang, Yayi Yang, Chongyang Shi, An Lao, Liang Hu, Shoujin Wang, and Usman Naseem

**Abstract**—The rapid growth of social media has caused tremendous effects on information propagation, raising extreme challenges in detecting rumors. Existing rumor detection methods typically exploit the reposting propagation of a rumor candidate for detection via regarding all reposts to a rumor candidate as a temporal sequence and learning semantics representations of the repost sequence. However, it is crucial to mine informative support from the topological structure of propagation and the influence of reposting authors for debunking rumors, which generally has not been well addressed by existing methods. In this paper, we organize a claim post in circulation as an adhoc event tree, extract event elements, and convert it to bipartite adhoc event trees in terms of both posts and authors, i.e., author tree and post tree. Accordingly, we propose a novel rumor detection model with hierarchical representation on the bipartite adhoc event trees called BAET. Specifically, we introduce word embedding and feature encoder for the author and post tree, respectively, and design a root-aware attention module to perform node representation. Then we adopt the tree-like RNN model to capture the structural correlations and propose a tree-aware attention module to learn tree representation for the author tree and post tree, respectively. Extensive experimental results on two public Twitter datasets demonstrate the effectiveness of BAET in exploring and exploiting the rumor propagation structure and the superior detection performance of BAET over state-of-the-art baseline methods.

**Index Terms**—Rumor detection, hierarchical representation, attention networks, neural networks

## I. INTRODUCTION

WITH the rapid development of social platforms, social media has become a convenient and essential tool for people to release and access news, comments, and speak out. However, social media can be a double-edged sword, as it facilitates the release and rampant spread of large amounts of fake or unverified information, leading to tremendous adverse effects on our society [1], [2]. For example, widespread rumor-mongering has seriously impacted political decision-making and manipulated public opinions in the 2016 US presidential election [3]. During the COVID-19 public health crisis, the wide-ranging rumor that oral disinfectants can prevent

Qi Zhang is with Data Science Lab, the University of Technology Sydney, Ultimo, NSW 2007, Australia and Beijing Institute of Technology, Beijing, 100081, China. E-mail: qi.zhang-13@student.uts.edu.au

Yayi Yang, Chongyang Shi (corresponding author), and An Lao are with the School of Computer Science, Beijing Institute of Technology, Beijing, 100081, China. E-mail: {yayi.yang, cy\_shi, an.lao}@bit.edu.cn

Liang Hu is with Tongji University, Shanghai, 200092, China and DeepBlue Academy of Sciences, Shanghai, 200336, China. E-mail: rainmilk@gmail.com

Shoujin Wang is with Data Science Institute, the University of Technology Sydney, Sydney, NSW 2007, Australia. E-mail: shoujin.wang@uts.edu.au

Usman Naseem is with the University of Sydney, Sydney, Camperdown, NSW 2006, Australia. E-mail: usman.naseem@sydney.edu.au

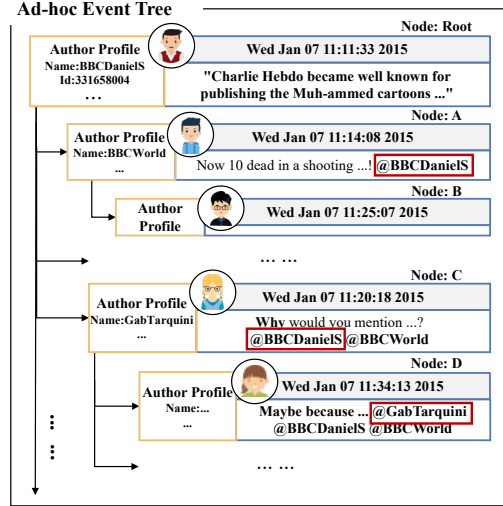


Fig. 1. The tree shows an adhoc event in circulation which is derived from the claim post *Charlie Hebdo headquarters being attacked by armed men*. In the tree, each node denotes a tuple related to the event, including a post related to the event, author profile, timestamp of the post. Each arrow denotes one repost operation. The root node denotes the claim post and all the others denote reposted posts. The “@” in the red box forms the repost.

infection and that heating masks can kill the virus led to the emergence of a large-scale health crisis [4]. Thus, detecting rumors is urgent and beneficial to avoid their potential threat and negative influences on social media and society.

The sociological definition of a rumor is a circulating piece of unverified information (e.g., post) [5], [6]. The task of rumor detection aims to verify the information veracity and distinguish rumors from non-rumors [7]. Dominant methods treat rumor detection as a binary classification task and solve the task referring to the three key properties of rumors, i.e., post content, author and temporal propagation [5]. For example, traditional methods adopt handcrafted features of text contents, user characteristics, and propagation patterns to train a classifier based on decision tree [8] or support vector machine [9]. Deep learning methods mine semantic features of stance or attitude from post contents [10], [7], [11] or capture high-level representations of sequential features and structural information from propagation paths [12], [13] to make the classification.

Both traditional and deep learning methods conclude that the temporal propagation of a rumor, in addition to its inherent information (i.e., its post and author), is effective for verifying the veracity of the rumor. In addition, increasingly prevailing methods pay more attention to the auxiliary information along

with propagation and learn informative representation from the rumor propagation structure for rumor detection. These methods can be roughly divided into two categories: (1) methods modeling rumor propagation from the chronological order perspective, and (2) methods modeling rumor propagation from a topological perspective. For example, RNN-based models [14], [15] arrange claim posts and comments in chronological order and exploit the sequential texts using recurrent neural units and attention mechanism for rumor detection. These models simplify posts' propagation and ignore the topological structure in the propagation. To bridge this gap, some traditional models, e.g., graph-kernel based methods [16], [17], and recent neural models, e.g., tree-based RvNN [18] and graph-based Bi-GCN [19], [20], exploit the topological structure of rumor propagation to learn informative representations for rumor detection. Notably, these methods benefit from the chronological order and topological structure of rumor propagation and thus obtain more effective information to debunk rumors. This indicates that the structure information of rumor propagation implies important spreading behaviors of rumors.

However, the above methods generally model the structure of the rumor propagation path from the perspective of posts while ignoring authors during rumor propagation. Intuitively, the veracity of a rumor depends on not only its post content but also its propagation information, including the semantics (e.g., stance and attitude) of posts and the influence of (re-posting) authors (e.g., reputation and reliability). Quite a few existing methods leverage authors' influence to enhance rumor detection. These methods [21], [22] typically treat author information as additional information to the post content to improve the presentation of the text content of the post. Considering the authors' influence, we treat a claim post of a rumor or non-rumor in circulation as an adhoc event, including the authors (who), the time sequence (when), and the posts (what) in circulation. Each adhoc event is organized with a tree to describe the propagation path of the information, which is denoted as an adhoc event tree as shown in the left on Fig. 1. We argue that author information, e.g., profiles, which implies his/her own reliability, also play an indispensable role in judging the veracity of his/her claims. For example, a verified author is indicated by the attribute "verified", and an author with a large number of followers may have a high probability of posting a true claim. To this end, it is necessary and reasonable to exploit author information in the adhoc event to judge the veracity of the corresponding posts. In addition, as the saying goes that "birds of a feather flock together", it is intuitively effective to exploit the information of authors in circulation to verify the veracity of their posts and support the reliability of the author of the claim post.

In light of the above discussion, we convert the adhoc event tree as shown in Fig. 1 into bipartite adhoc event trees in terms of both posts and authors, denoted as *author tree* and *post tree* respectively, as shown in Fig. 2. Based on the obtained bipartite trees, the major challenge for debunking rumors is to obtain sufficient information on both posts and authors from the post tree and author tree, respectively. This triggers two challenges (a) how to learn an informative representation for

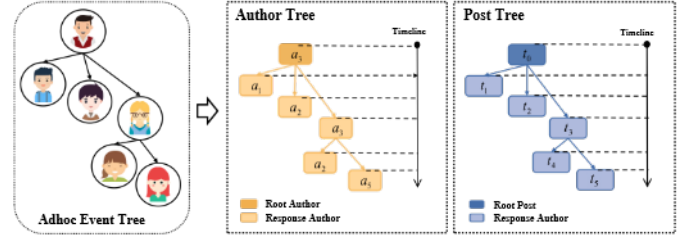


Fig. 2. The propagation structure of the adhoc event tree (left) in Fig. 1 and the corresponding converted bipartite adhoc event trees (right). The author (post) tree has the same structure as the original adhoc event tree. In the author (post) tree, each node denotes an author (post), and all nodes are arranged in chronological order from top to bottom.

each of the tree nodes, and (b) how to effectively capture the structural correlations between the root post (author) and the response posts (authors) within the bipartite adhoc trees to represent the propagation structure? Accordingly, we propose a novel rumor detection model which learns hierarchical representation from the bipartite adhoc event trees called BAET. The model introduces word embedding and feature encoder for the author and post tree, respectively, to learn the embedding of posts and authors, and a root-aware attention module to learn node representation for both posts and authors to address the challenge (a). Motivated by [18], we further introduce the tree-like RNN model (i.e., RvNN) to capture the aforementioned structural correlations and propose a tree-aware attention module to learn tree representation for the author tree and post tree respectively to address the challenge (b). Our contributions can be summarized as follows:

- We treat a claim post in circulation as an adhoc event from which we construct bipartite (post/author) adhoc trees to provide additional author information for rumor detection. To achieve this, we propose a novel rumor detection model called BAET, utilizing signals from both posts and authors to debunk rumours effectively.
- We introduce word embedding and feature encoder to embed post and author, respectively, and design a root-aware attention module for learning node-level representation effectively.
- We adopt RvNN to learn the structural correlations over nodes within trees and then propose a tree-aware attention module to adaptively learn the structural-level representation from the bipartite adhoc trees.
- Extensive experimental results show that BAET achieves significant improvements compared with the state-of-the-art rumor detection models on public rumor detection datasets PHEME and RumorEval, which proves the contribution of author information to rumor detection.

## II. RELATED WORK

### A. Early rumor detection based on feature engineering

Researchers in different disciplines have investigated rumor detection, including philosophy and humanities [23], social psychology [24], [25], computer science and artificial intelligence [26], [27]. It is difficult to identify rumors early on because a large number of news investigations are required to

verify suspicious tweets. With the increased usage of social media, there is a massive information overload, making rumor detection a time-consuming and labor-intensive task. Early studies focused on using manual linguistic features extracted from rumor textual contents (source/root and responsive posts) to decide rumor labels (i.e., true or false) via traditional machine learning methods [28], [29], [30]. For example, Zhao et al. [28] believed that people exposed to a rumor claim would seek more information or express skepticism before deciding whether to believe it or not. Thus, he clustered the claim and its successor texts which contain doubtful expressions such as “Is this true?”, “Really?”, and then used classical classification models such as SVM and decision tree to conduct the classification task. Ma et al. [17] aggregated all of the responses to the root claim to form a rumor propagation tree and then identified the claim using SVM with different kernel numbers.

These methods rely on manually created features, which can not guarantee the accuracy of rumor classification. To achieve real-time tracking and identification of rumors, researchers must shift their focus to automatic rumor detection methods. The initial research on the automatic rumor detection method is based on text mining with supervised models built on feature engineering. For instance, Ma et al. [31] proposed a dynamic time series structure (DSTS) that can capture a variety of social context text features based on time series changes, as well as the various features generated by rumors along with time, allowing rumors to be detected in different time periods. Rath et al. [32] proposed a new trustworthiness concept based on responsive post characteristics to identify authors who spread rumors automatically. Because of the large amount of noise information, such feature engineering-based methods extract more single or coarse-grained features, and the accuracy of the classification task cannot be guaranteed.

### B. Automatic rumor detection based on deep learning

Deep learning research solves the automatic feature extraction problem while achieving fine granularity and diversity. Researchers are no longer limited to the single processing and use of features because of richer and more accurate feature information. However, they are instead exploring the fusion of multiple features and structural modeling.

**Serialized propagation topology representation.** Some methods place less emphasis on the propagation topology and instead focus on propagation time sequence properties. These researchers hoped to explore relevant clues generated over time to characterize the different characteristics of the rumor at different times by modeling the propagation process of information with time characteristics. Yu et al. [33] designed a CNN-based model called CAMI to extract flexible content sequence between key characteristics, formation of high-level interactions, and aids in the effective classification. Extensive mixed features, such as time series and content-related features, are becoming increasingly important to research. Thus Ma et al. [14] used a RNN network to process each time step in the rumor propagation sequentially. Chen et al. [15] proposed an attention mechanism to capture longer-term dependencies

and enhance the long-term temporal features [34]. In addition, attention networks (CNN) [ ] and Transformers [35] were adopted to mine the posts better features with time sequences as a clue to identify rumors.

Author traits, on the other hand, are frequently used to define authors’ personality information, such as authors’ profile information and social relationships, which are used to vividly describe the authors involvement in propagation and their behavior [16]. Shu et al. [36] argued that fake news could imitate real news, which means that the dissemination of authors’ information must be thoroughly researched. Ruchansky et al. [37] developed a tri-relationship embedding framework TriFN that can learn the characteristics of forwarded texts and author profiles while also generating credibility scores to help with rumor detection. Shu et al. [38] built five novel metrics to interactively model the relationships between news and users (publishers, social participants, etc.). Recently, Wu et al. [21] constructed a decision tree mechanism to address rumor interpretability issues, using users’ credibility scores as evidence to help identify rumors. Lu et al. [22] built a graph-aware joint attention network to detect fake news by taking into account the root article’s text content (claim), forwarding author sequence (sequential propagation) and author information (author profile). These methods, however, cannot accurately reflect the characteristics of actual rumor propagation and spreading because they focus on mining the characteristics of authors and posts changing over time via simple temporality while ignoring the internal topology of rumor propagation.

**Structured propagation topology representation.** The rumor propagation topology representation is concerned with modeling the rumor propagation structure, gathering various information such as stances and sentiments in the topology [?], and mining the interactive information of multiple features of posts and authors to enrich the rumor representation. Simultaneously, understanding the process and mode of rumor propagation aids in identifying rumors at an early stage of rumor propagation and achieves a good explanatory effect in detecting the authenticity of rumors. Rumor propagation topology enables us to better abstract rumor propagation patterns and further excavate the correlations between hybrid features via propagation aggregation [39], [40], resulting in good interpretability on the veracity of rumors. Wu et al. [16] calculated the substructural similarity of two propagation trees on Sina Weibo datasets for detection. Similarly, Ma et al. [17] employed the tree kernel to compute similar substructures and identify various types of rumors in Twitter datasets. The above two methods rely on the similarity of the two trees. However, they cannot distinguish between individual trees and are still based on feature engineering rather than deep learning. Moreover, considering that RvNN [41], [42] were originally used to learn phrase or sentence representations for semantic parsing, Wong et al. [18] proposed two types of recursive tree-structured neural network models to learn representations from their structure and contents. Ma et al. [43] recently developed an improved rumor detection model based on Wong et al. [18] with designed discriminative attention mechanisms for model optimization.

Bian et al. [19] believed that Wong et al. [18] only considered propagation depth, ignoring spread. Two graph convolutional networks (GCN) were used to learn the top-down propagation depth and the down-top dispersed breadth simultaneously. Wei et al. [20] is also based on GCN networks and considers the reliability of the rumor propagation relationship when optimizing the propagation model for a better rumor classification effect. Graph attention networks (GAT) are also used in various research fields. Lin et al. [44] proposed a new ClaHi-GAT model to represent the action graph between the post content and the author and learn the characteristics of rumors by the multi-layer attention mechanism of the root post layer and event layer. In addition, Liu et al. [45] proposed a novel rumor detection framework based on structure-aware retweeting GNNs based on a converted tractable tree. Wei et al. [46] further considered the uncertainty of interactions caused by users' various subjective factors and proposed fuzzy graph convolutional networks to model uncertain interactions in the information cascade of a tweet graph. Although such methods are constantly improving the modeling of rumor propagation topological structure to obtain better rumor representations, most ignore author-level information, including simple user personality characteristics (author profiles) and the structured interaction relationship formed between users during the spread process.

### III. PROBLEM STATEMENT

In this section, we introduce the problem statement of rumor detection, of which the task is to learn a classifier to distinguish a claim post of a rumor or non-rumor. To achieve this, we treat the claim post with its responsive posts in circulation as an adhoc event tree and build bipartite adhoc event trees consisting of a post tree and an author tree correspondingly to extract more informative supports for detection. Formally, let  $\Gamma = \{t_0, t_1, t_2, \dots, t_n\}$  be a sequence of posts in a post tree in chronological order where  $n$  denotes the number of responsive posts. Herein,  $t_0$  corresponding to a claim post is the root node of the post tree,  $t_k (k > 0)$  in  $\Gamma$  corresponds to the  $k$ -th responsive posts, and  $\{t_1, t_2, \dots, t_n\}$  denotes the sequence of re-posts. Correspondingly, we denote the sequence of authors in an author tree as  $\Theta = \{a_0, a_1, a_2, \dots, a_n\}$ , where each author  $a_i$  published the claim  $t_i$  and is involved with various features, such as the number of followers and friends, as shown in Table II. We further build the responsive connections between posts (authors) within the post (author) sequence from the "forward" and "reply" operations revealed by the sign "@" in the posts, depicting the propagation structure of the post (author) tree. Accordingly, a classifier  $C$  is then build to predict the label  $y$  of the claim  $t_0$  associated with the inputs of  $\Gamma$  and  $\Theta$  where  $y \in \{0, 1\}$  and 0 indicates a rumor otherwise 1, that is  $C : (t_0, \Gamma, \Theta) \rightarrow y$ . Allowing for obtaining propagation information, we revisit the critical edge-directional propagation problem in the tree and treat the direction of each arrowed connect between posts (authors) indicating the direction of information transmission.

TABLE I  
PRIMARY NOTATION DESCRIPTIONS

Notation	Explanation
$t_0$	The claim post text
$t_i$	The $i$ -th responsive post text
$a_0$	The claim author
$a_i$	The $i$ -th responsive author
$\mathbf{E}_i$	The embedding matrix of $i$ -th post text
$\mathbf{F}_i$	The feature representation matrix of $i$ -th author
$\mathbf{h}_i$	The $i$ -th hidden representations of GRU Cell

### IV. THE BAET MODEL

We introduce our rumor detection model BAET in this section. The overall architecture of BAET is depicted in Fig. 3, containing three key components: *Node-level Representation Module*, *Structural-level Representation Module*, and *Prediction Module*. Table I summarizes the key notations that are used throughout this paper.

#### A. Node-level Representation Module

We employ the node-level representation module in both the post and author trees to learn a more informative representation of each node from the perspectives of semantics (content) and interactive (propagation) correlations in the bipartite tree. Specifically, this module consists of two sub-modules: *Tree Nodes Process* (TNP) and *Root-aware Attention Learning* (RAL). TNP is introduced to perform embeddings of post nodes and author nodes. The embeddings are then fed into the RAL model to learn the correlations between words/author features within each post/author and the semantic/interactive correlations between each responsive post/author and the claim (root) post/author.

**(1) Tree Node Process (TNP).** We first introduce *Tree Node Process* in the post tree to capture the text information of all posts in an adhoc event. The topological post tree obtained in Fig. 2 is processed along the tree structure, where all post text information and topological structure in the tree are recorded. Notably, the topological structure in the tree mainly consists of root nodes, non-leaf nodes, and leaf nodes. We denote the maximum length of the post text as  $L_m$  and collect all words in all post texts as a corpus. All words are then tokenized such that a corpus index can represent each word. Formally, all the tokenized texts are initialized as  $\{\tilde{\mathbf{t}}_0, \tilde{\mathbf{t}}_1, \tilde{\mathbf{t}}_2, \dots, \tilde{\mathbf{t}}_n\}$  with  $\tilde{\mathbf{t}}_i \in \mathbb{R}^{L_m \times 1}$  and mapped into a low-dimensional space through an embedding layer as follows:

$$\tilde{\mathbf{E}}_i = \text{Embedding}(\tilde{\mathbf{t}}_i) \quad (1)$$

where  $\tilde{\mathbf{E}}_i \in \mathbb{R}^{L_m \times d}$ ,  $i \in [0, n]$ , and  $d$  is the dimension of embedding. We further count the frequency of each word appearing in its post and obtain the frequency vectors of all posts as  $\{\mathbf{q}_0, \mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n\}$  with the word frequencies of  $i$ -th post  $\mathbf{q}_i \in \mathbb{R}^{1 \times L_m}$  and  $i \in [0, n]$ . Then, we get the frequency-enhanced embeddings of posts as below:

$$\mathbf{E}_i = \mathbf{q}_i * \tilde{\mathbf{E}}_i \quad (2)$$

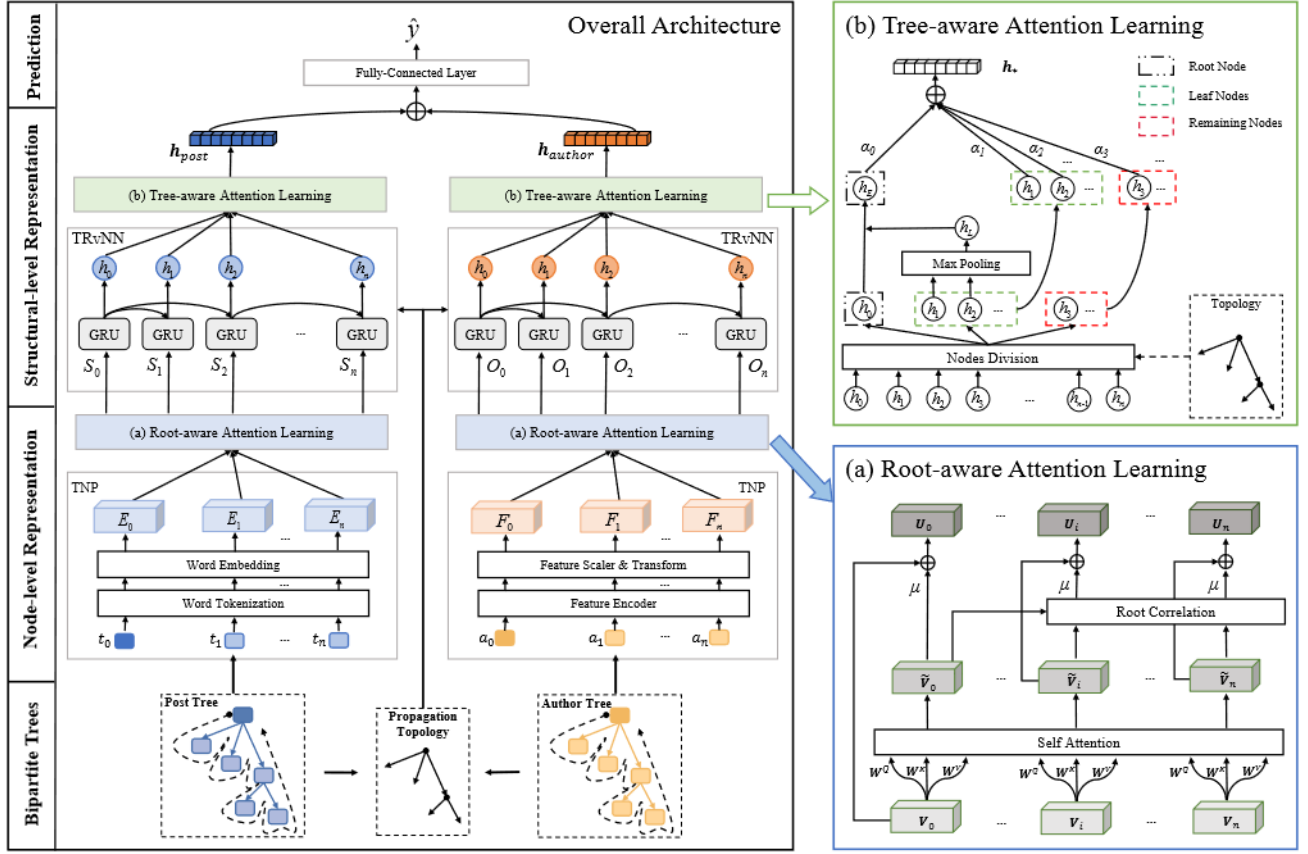


Fig. 3. The overall architecture of BAET. It comprises three key modules: *Node-level Representation*, *Structural-level Representation*, and *Prediction* modules. 1) Node-level representation module includes two sub-modules of *Tree Node Process* (TNP) and *Root-aware Attention Learning* (TAL). 2) Structural-level representation module contains two sub-modules of *Tree Variant RNN* (TRvNN) and *Tree-aware Attention Learning* (TAL). 3) Prediction layers are adopted to classify rumors and non-rumors. Both node-level representation modules and structure-level representation modules are applied to the post tree and author tree to learn more informative representations of post nodes and author nodes, respectively.

TABLE II  
DETAILS OF AUTHOR FEATURES

Category	Description	Value Range
Author basic features	number of followers	int [0, $\infty$ )
	number of friends	int [0, $\infty$ )
	number of favorites	int [0, $\infty$ )
	number of reposts	int [0, $\infty$ )
	number of statuses	int [0, $\infty$ )
	verified	bool (0/1)
	geo-enabled	bool (0/1)
	time-zone enabled	bool (0/1)
	time of posting	timestamp
	time of creating an account	timestamp
Author writing-habit features	similarity to claim	float [0, 1)
	post text length	int [0, 30]
	max text length (a tree)	int [0, 30]
	average word length	int [0, 20]
	question mark count	int [0, 10]
	exclamation mark count	int [0, 10]

where  $*$  denotes the element-wise product, and  $\mathbf{E}_i \in \mathbb{R}^{L_m \times d}$ . Accordingly, we obtain the embeddings of all the posts in the post tree:  $\{\mathbf{E}_0, \mathbf{E}_1, \dots, \mathbf{E}_n\}$ .

Regarding the author nodes, we have two types of author features, i.e., the author's basic features and the authors'

writing habits, as shown in Table II. Specifically, “similarity to claim” is calculated using Jaccard similarity, expressing the relevance between the repost and the claim post, “post text length” records the number of words in the post, “max post text length” is the maximum of text length in the tree, and “average word length” reveals the author's habits in terms of the average length of words in his/her posts.

We perform feature encoding for each type of feature separately and then adopt one-hot encoding to transform categorical features such as “verified”, “geo-enabled”, and “time-zone enabled”. In addition, the timestamp features (i.e., “time of posting” and “time of creating an account” are replaced with timestamp intervals calculated as follows:

$$\Delta s_i = \log |s_i - s_0 + 1|, \quad i \in [0, n] \quad (3)$$

where  $s_0$  and  $s_i$  correspond to the timestamps of the root post and the  $i$ -th responsive post, respectively. Accordingly, we obtain two matrices of raw basic features and writing-habit features for all authors. The two matrices are then normalized to  $[0, 1]$  using the min-max normalization separately, resulting in two normalized feature matrices  $\mathbf{F}^b = [\mathbf{f}_0^b, \mathbf{f}_1^b, \dots, \mathbf{f}_n^b]$  and  $\mathbf{F}^h = [\mathbf{f}_0^h, \mathbf{f}_1^h, \dots, \mathbf{f}_n^h]$  where  $\mathbf{f}_i^b$  and  $\mathbf{f}_i^h$  denote the normalized feature vectors of the basic features and writing-habit features



of the  $i$ -th author (i.e.,  $a_i$ ), respectively. Finally, we can obtain the feature representation for each author calculated as follows:

$$\mathbf{F}_i = \mathbf{f}_i^b \mathbf{W}^b \oplus \mathbf{f}_i^h \mathbf{W}^h, \quad i \in [0, n] \quad (4)$$

where  $\mathbf{F}_i \in \mathbb{R}(|\mathbf{f}_i^b| + |\mathbf{f}_i^h|) \times d$  denotes the representation embedding of author  $a_i$ ,  $\mathbf{W}^b \in \mathbb{R}^{1 \times d}$  and  $\mathbf{W}^h \in \mathbb{R}^{1 \times d}$  denote the transform matrices, and  $\oplus$  denotes the concatenation operation. Finally, we obtain the feature representations of all authors  $\{\mathbf{F}_0, \mathbf{F}_1, \dots, \mathbf{F}_n\}$ .

**(2) Root-aware Attention Learning (RAL).** We propose a root-aware attention learning module consisting of two attention modules to learn comprehensive node representations for both the post and author trees. As shown in Fig. 3 (a), we use  $\{\mathbf{V}_0, \mathbf{V}_1, \dots, \mathbf{V}_n\}$  to denote the representation matrices obtained from either the post tree or author tree via the tree nodes process module, i.e.,  $\mathbf{V}_i = \mathbf{E}_i$  or  $\mathbf{V}_i = \mathbf{F}_i$ .

First, to learn the correlations between words (features) within a post (author) nodes, we leverage the self-attention mechanisms to enhance the representations obtained through the *Tree Nodes Process* module to capture the semantics or features correlations within each post/author node:

$$\tilde{\mathbf{V}}_i = \mathcal{A}(\mathbf{V}_i \mathbf{W}^Q, \mathbf{V}_i \mathbf{W}^K, \mathbf{V}_i \mathbf{W}^V) \quad (5)$$

where  $\mathbf{W}^Q, \mathbf{W}^K, \mathbf{W}^V \in \mathbb{R}^{d \times d}$  are projection weight parameters,  $\mathcal{A}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V}$  denotes the calculation of self-attention, and  $\tilde{\mathbf{V}}_i$  denotes the output of self-attention.

Secondly, with the increase in the number of participants in actual rumor/non-rumor propagation, the claim post information and its author's characteristic information in the root node may be weakened or forgotten in circulation. To address this, we introduce another root-aware attention module to learn the interactive correlations between the root node and each responsive node:

$$\hat{\mathbf{V}}_i = \text{Sigmoid}([\tilde{\mathbf{V}}_0, \tilde{\mathbf{V}}_i] \mathbf{W}^a + \mathbf{b}^a) * \tilde{\mathbf{V}}_i \quad (6)$$

where  $\mathbf{W}^a \in \mathbb{R}^{2d \times d}$  are projection weights,  $\mathbf{b}^a \in \mathbb{R}^d$  is the bias term,  $\tilde{\mathbf{V}}_0$  and  $\tilde{\mathbf{V}}_i$  are the self-attention representations of the root node and the  $i$ -th responsive node, respectively, and  $\hat{\mathbf{V}}_i$  is the attention output attending to both the two nodes.

Finally, we fuse the outputs of the two attention modules:

$$\mathbf{U}_0 = \tilde{\mathbf{V}}_0 + (\mu * \mathbf{V}_0) \quad (7)$$

$$\mathbf{U}_i = \tilde{\mathbf{V}}_i + (\mu * \hat{\mathbf{V}}_i) \quad (8)$$

where  $\mathbf{U}_0$  and  $\mathbf{U}_i$  denote the fused representations of the root node and the  $i$ -th responsive node, respectively. In addition,  $\mu$  is a hyper-parameter used to determine the extent of how much interactive correlation between the root node and responsive node should be considered. Subsequently, substituting  $\mathbf{V}_i = \mathbf{E}_i$  or  $\mathbf{V}_i = \mathbf{F}_i$  to the above Equations 5-8 separately, we can obtain the representations of all post nodes and author nodes in the bipartite trees which are denoted as  $\{\mathbf{S}_0, \mathbf{S}_1, \dots, \mathbf{S}_n\}$  and  $\{\mathbf{O}_0, \mathbf{O}_1, \dots, \mathbf{O}_n\}$ , respectively.

### B. Structural-level Representation Module

All node-level representations learned above are fed into the *Structural-level Representation module* to capture a comprehensive representation of both the post tree and author

tree. The module consists of two sub-modules: *Tree Variant RNN* (TRvNN) and *Tree-aware Attention Learning* (TAL). TRvNN is a structure-aware GRU layer that simulates the hidden transitions according to the connections in the tree, while TAL aggregates the output hidden states from TRvNN to generate the representation vectors via a tree-aware attention mechanism.

**(1) Tree Variant RNN (TRvNN).** We incorporate the propagation tree topology to RNN to form TRvNN, a tree-variant RNN inspired by [18]. We choose GRU cells with one node corresponding to one GRU cell to guarantee computation efficiency in modeling the long propagation tree.

Specifically,  $\{\mathbf{S}_0, \mathbf{S}_1, \dots, \mathbf{S}_n\}$  and  $\{\mathbf{O}_0, \mathbf{O}_1, \dots, \mathbf{O}_n\}$  are served as inputs to TRvNN. We assume that all information from a non-leaf node in the post tree or author tree can be passed to all of its descendant node. Therefore, the hidden state of the GRU cell is calculated by combining the representations of the current node and the hidden state of its direct parent node. Formally, given all node representations of the post tree or author tree  $\{\mathbf{U}_0, \mathbf{U}_1, \dots, \mathbf{U}_n\}$ , i.e.,  $\mathbf{U}_i = \mathbf{S}_i$  or  $\mathbf{U}_i = \mathbf{O}_i$ , the GRU cell in our TRvNN is calculated as follows:

$$\begin{aligned} \tilde{\mathbf{U}}_i &= \mathbf{U}_i \mathbf{W}^U + \mathbf{b}^U \\ \mathbf{r}_i &= \sigma_r([\tilde{\mathbf{U}}_i, \mathbf{h}_{P(i)}] \mathbf{W}^r) \\ \mathbf{z}_i &= \sigma_z([\tilde{\mathbf{U}}_i, \mathbf{h}_{P(i)}] \mathbf{W}^z) \\ \tilde{\mathbf{h}}_i &= \tanh([\tilde{\mathbf{U}}_i, \mathbf{h}_{P(i)} * \mathbf{r}_i] \mathbf{W}^H) \\ \mathbf{h}_i &= (1 - \mathbf{z}_i) * \mathbf{h}_{P(i)} + \mathbf{z}_i * \tilde{\mathbf{h}}_i \end{aligned} \quad (9)$$

where  $P(i)$  is the parent of the  $i$ -th node in either the post tree or author tree,  $\sigma_r, \sigma_z$  are the activation functions (precisely Sigmoid used in the experiments),  $\mathbf{W}^U \in \mathbb{R}^{d \times d}$  and  $\mathbf{b}^U \in \mathbb{R}^d$  denote the transformation weights and the bias term, respectively, and  $\mathbf{W}^r, \mathbf{W}^z, \mathbf{W}^H \in \mathbb{R}^{2d \times d}$  are the projection weights for calculating the reset gate  $\mathbf{r}_i$ , the update gate  $\mathbf{z}_i$  and the candidate activation  $\tilde{\mathbf{h}}_i$ , respectively. Accordingly, we can obtain all output hidden state  $\{\mathbf{h}_i\}_{i=0}^n$  from TRvNN, resulting in  $\{\mathbf{h}_i^p\}_{i=0}^n$  and  $\{\mathbf{h}_i^a\}_{i=0}^n$  corresponding to all post nodes and author nodes, respectively.

**(1) Tree-aware Attention Learning (TAL).** To aggregate the output hidden state from TRvNN and capture structural information from the post/author tree, we propose a tree-aware attention learning module. This module leverages attention mechanisms and adopts  $\{\mathbf{h}_i^p\}_{i=0}^n$  and  $\{\mathbf{h}_i^a\}_{i=0}^n$  as inputs to obtain the final structural representations of the post tree and author tree. For simplicity, we introduce the module using  $\{\mathbf{h}_i\}_{i=0}^n$ , as shown in Fig. 3(b).

First, based on the topological structure, we divide the obtained hidden state representations into three categories: the hidden states of the root node, leaf nodes, and other nodes, which are denoted by dotted boxes of different colors as shown in Fig. 3(b). The treatment aims to show that different kinds of nodes in the post tree or author tree contribute differently to spreading rumors. That is, we distribute different weights to different kinds of nodes to learn tree-aware representations of the post/author tree:

(a) *The leaf nodes* reflect the information aggregation process from the root node to the current leaf nodes along

with the tree structure. Intuitively, the emotional intensity and propagation flow in the path is all gathered in the leaf nodes.

(b) *The other nodes* act on the paths from the root node to the leaf nodes. Their node representations in the current path may influence the other paths. For example, the negative emotion in the current node may affect the expression of the nodes on other paths.

(c) *The root node* contains the most important information in the propagation. However, during recursive learning, the information in the root node may be weakened or forgotten as the path length increases.

Following the above node division, all leaf nodes are fed into a max-pooling layer to obtain the maximum value of each vector element of all leaf nodes, thereby representing the representation of all leaf nodes. Intuitively, this treatment can account for and capture the most attractive and indicative representation of all propagation paths:

$$\mathbf{h}_L = \text{MaxPooling}(\mathbf{h}_{l1}, \mathbf{h}_{l2}, \mathbf{h}_{l3} \cdots) \quad (10)$$

where  $\{\mathbf{h}_{l1}, \mathbf{h}_{l2}, \mathbf{h}_{l3}, \dots\}$  denote the hidden states of the leaf nodes, and the resultant  $\mathbf{h}_L \in \mathbb{R}^d$  denotes the maximum representation on all the leaf nodes. Accordingly, we calculate the correlation between this maximum representation and the root node representation and obtain an enhanced representation for the root node  $\mathbf{h}_E$ :

$$\mathbf{h}_E = \sigma_E([\mathbf{h}_0, \mathbf{h}_L]\mathbf{W}^E + \mathbf{b}^E) \quad (11)$$

where  $\mathbf{h}_0$  denote the hidden state representation of the root node,  $\sigma_E$  is the activation function (tanh used in the experiments),  $\mathbf{W}^E \in \mathbb{R}^{2d \times d}$  and  $\mathbf{b}^E \in \mathbb{R}^d$  are the weight parameters and bias term, respectively. On the one hand, root node information can be fused with indicative maximum leaf node information to perform a strong combination via Equation 11. On the other hand, it can compensate for the loss of root node information due to a long propagation.

Subsequently, the representation  $\mathbf{h}_E$ , along with other node hidden states  $\{\mathbf{h}_i\}_{i=1}^n$ , will be fed into a self-attention network. We expect to learn their correlations and capture the contribution of each node to the final tree structure representation. Let  $\mathbf{h}_i$  be the  $i$ -th node hidden states, where  $i \in [0, n]$  and  $\mathbf{h}_0 = \mathbf{h}_E$ . We calculate the attentive weight  $\alpha_i$  for each  $\mathbf{h}_i$ :

$$\alpha_i = \frac{\exp(\mathbf{h}_i \mathbf{w}_c^T)}{\sum_{j=0}^n \exp(\mathbf{h}_j \mathbf{w}_c^T)}, \quad (12)$$

where  $\mathbf{w}_c \in \mathbb{R}^d$  is the weight parameters. Accordingly, we obtain the representation for the post tree or author tree via the attentive weighted summation:

$$\mathbf{h}_* = \sum_{i=0}^n \alpha_i \mathbf{h}_i \quad (13)$$

Accordingly, we can obtain  $\mathbf{h}_{post}$  and  $\mathbf{h}_{author}$  according to Equation 13 for the post tree and author tree, respectively.

### C. Prediction Layers and Loss Function

To obtain comprehensive representations, we concatenate the representations of the post tree  $\mathbf{h}_{post}$  and author tree  $\mathbf{h}_{author}$ , fusing the shared features of the two trees:

$$\mathbf{h}_o = \mathbf{h}_{post} \oplus \mathbf{h}_{author} \quad (14)$$

TABLE III  
STATISTICS OF THE DATASETS

Statistic	PHEME	RumorEval
Number of claim posts	3,589	297
Number of authors	57,961	4,261
Number of all posts	61,782	4,841
Number of rumors	1,803	137
Number of non-rumors	1,786	160
Average number of reposts	16.2	15.3
Average tree depth	4.6	4.1

where  $\mathbf{h}_o \in \mathbb{R}^{2d}$  denotes the output representation for prediction. Since we define the rumor detection task as a binary classification problem, we utilize a binary label  $y \in [0, 1]$  as the ground truth. To predict the label of the claim (the root node of the post tree), we feed  $\mathbf{h}_o$  into a full-connection layer with the Softmax activation as follows:

$$\hat{y} = \text{Softmax}(\mathbf{h}_o \mathbf{W}^y + \mathbf{b}^y) \quad (15)$$

where  $\mathbf{W}^y \in \mathbb{R}^{2d \times 2}$ ,  $\mathbf{b}^y \in \mathbb{R}^2$  are parameter weights and the bias term. Our model is trained to minimize the cross-entropy loss between each prediction  $\hat{y}$  and the ground truth  $y$  as below:

$$\mathcal{L}(\Theta, \Omega) = - \sum [y \log \hat{y} + (1 - y) \log(1 - \hat{y})] \quad (16)$$

where  $\mathcal{L}$  denotes the cross-entropy loss,  $\Theta = \{\{\mathbf{W}\}, \{\mathbf{b}\}\}$  denotes the set of trainable weight parameters, and  $\Omega = \{d, \mu\}$  is the set of hyperparameters.

## V. EXPERIMENTS

### A. Experiment Details

**Datasets.** We adopt two publicly available datasets, i.e., *PHEME*<sup>1</sup> [47] and *RumorEval*<sup>2</sup> [48], for evaluation. Detailed statistics of these two datasets are provided in Table III. Both datasets are real-world data collected from the social platform Twitter, and the RumorEval dataset was developed for the SemEval-2017 Task 8 competition. The PHEME dataset contains five breaking news events, while RumorEval involves eight breaking news events. Both datasets contain post texts, author information, timestamps, and propagation information. Each claim is labeled with a binary class (i.e., True Rumor or False Rumor). In addition, during data preprocessing of the PHEME data, we removed noisy samples that have less than 3 retweets or do not have author information associated with the authors (IDs) of the claimed post or retweet post. All comparative models are evaluated on the two datasets to provide an extensive and fair comparison.

**Baselines.** We conduct a comprehensive comparison of our model with state-of-the-art rumor detection baselines.

- *DTR* [28]: A decision tree-based ranking model proposed to identify trending rumors through searching query phrases.

<sup>1</sup>[https://figshare.com/articles/dataset/PHEME\\_dataset\\_of\\_rumours\\_and\\_non\\_rumours/4010619](https://figshare.com/articles/dataset/PHEME_dataset_of_rumours_and_non_rumours/4010619)

<sup>2</sup><https://alt.qcri.org/semeval2017/task8/index.php?id=data-and-tools>



TABLE IV  
PERFORMANCE COMPARISON AGAINST DIFFERENT BASELINES ON TWO DATASETS

Method	PHEME				RumorEval			
	Acc. (%)	Prec. (%)	Rec. (%)	F1 (%)	Acc. (%)	Prec. (%)	Rec. (%)	F1 (%)
DTR	58.30	58.56	58.07	58.25	56.10	53.66	61.97	57.52
CNN	59.23	56.14	64.64	60.09	61.90	54.54	66.67	59.88
TE	65.22	63.05	64.64	63.83	66.67	60.00	66.67	63.15
SVM	72.18	72.80	75.74	74.24	71.42	66.61	66.73	66.67
RvNN	81.75	81.24	82.46	81.85	81.48	78.49	85.32	81.77
DTCA	82.46	79.08	86.24	82.50	82.54	78.25	85.60	81.76
RvNN-GA	85.19	84.93	86.21	85.56	86.03	85.08	86.51	85.79
BiGCN	85.93	85.11	87.04	86.06	86.75	<u>87.79</u>	85.73	86.75
ClaHi-GAT	85.90	-	-	84.15	-	-	-	-
RGNN	86.17	85.98	86.29	86.57	87.03	86.87	87.51	86.79
EBGCN	86.52	<u>86.85</u>	86.38	<u>86.61</u>	87.26	87.17	88.09	<u>87.63</u>
FGCN	86.69	86.74	<u>87.07</u>	86.56	<u>87.37</u>	87.08	<u>88.26</u>	87.49
BAET	<b>87.84</b>	<b>87.37</b>	<b>89.22</b>	<b>88.29</b>	<b>88.50</b>	<b>88.13</b>	<b>89.46</b>	<b>88.79</b>

- *CNN* [30]: A convolutional neural network with multiple filter sizes that is used to detect the stance of posts and determine the veracity of specific rumors.
- *SVM* [31]: A linear SVM classifier used to identify rumors with leveraging temporal sequences to model the chronological variation of social context features.
- *TE* [49]: A content-based model that leverages tensor decomposition to derive concise article embeddings, capture contextual information and create an article-by-article graph on limited labels.
- *RvNN* [18]: A top-down and a down-top models based on tree-structured RvNN proposed to detect rumors through integrating structure and content. We choose the well-performed top-down model as our baseline.
- *DTCA* [21]: A decision tree-based co-attention model developed to detect rumors by discovering evidence for explainable verification, in which the evidence scores combine both authors' and texts' information.
- *RvNN-GA* [43]: An enhanced rumor detection model proposed with designed discriminative attention mechanisms based on TD-RvNN/BU-RvNN. In this work, we select the top-down global-attention model as a baseline.
- *Bi-GCN* [19]: A bi-directional graph model based on two layer GCNs to detect rumors through integrating structural features of both propagation depth (top-down) and dispersion (down-top).
- *ClaHi-GAT* [44]: A graph attention network model is proposed to represent posts and authors as an undirected graph and then augmented by a multi-layer attention mechanism to learn the rumor indicator characteristics.
- *EBGCN* [20]: An edge-enhanced rumor detection model proposed with multiple graph conventional layers to iteratively aggregate features of structural neighbors and take the uncertainty into consideration.
- *RGNN* [45]: A rumor detection framework based on structure-aware retweeting graph neural networks to integrate content, users, and propagation patterns from a converted tractable binary tree to enhance performance.
- *FGCN* [46]: A neuro-fuzzy method with fuzzy graph convolutional networks to capture uncertain interactions

in the information cascade in a fuzzy perspective based on the tree or tree-like structure of source tweets.

These representative and state-of-the-art baselines are chosen carefully for extensive comparisons. 1) DTR, CNN and SVM are representative baselines based on handcrafted features, and TE is a content-based detection model. Those baselines are compared to investigate the effectiveness of propagation information in the detection. 2) The other state-of-the-art baselines includes RNN-, attention-, and GNN-based models that elaborately leverage propagation information to enhance detection. Those baselines are selected to investigate the benefits of our proposed bipartite adhoc trees, especially the propagation information from the author perspective. In addition, multi-modality detection models are not considered in the experiments since those methods based on multi-modality information of text and image inputs do not belong to the family of propagation-based models.

**Experimental settings.** (1) *Parameter settings*: In the experiments, all parameters are initialized with a normal distribution, and we perform a grid search of the learning rate over  $\{1e^{-3}, 5e^{-3}, 1e^{-4}, 5e^{-4}, 1e^{-5}\}$  and L2 regularization rate over  $\{0, 1e^{-1}, 1e^{-2}, 1e^{-3}, 1e^{-4}\}$  to guarantee the best performance of our proposed BAET. For hyperparameters  $\Omega = d, \mu$ , we tune the dimension of the embedding and hidden units  $d$  over  $\{16, 32, 64, 128, 256\}$  and  $\mu$  over  $\{0.0, 0.2, 0.4, \dots, 1.0\}$  to choose the values with the best performance for BAET. Finally, we set i) the embedding dimension  $d = 128$ ; ii) a balance weight  $\mu = 0.6$ ; iii) the ADAM optimizer [50] with an initial learning rate of 0.005; iv) a batch size of 16; v) L2 regularization rate of  $1e^{-3}$ ; vi) a dropout rate of 0.5. (2) *Baseline settings*: Regarding the baselines, we implement RvNN, RvNN-GA, BiGCN, and EBGCN according to their published source codes, while other baselines are implemented by referring to the corresponding papers. To ensure fair comparisons, the best parameters for all comparative models are chosen by carefully tuning the author-recommended parameter ranges based on their recommended settings. Specifically, for the parameters in each baseline, we first initialize them with the values reported in the original paper and then carefully tune them on our datasets for best performance. (3) *Metrics*:

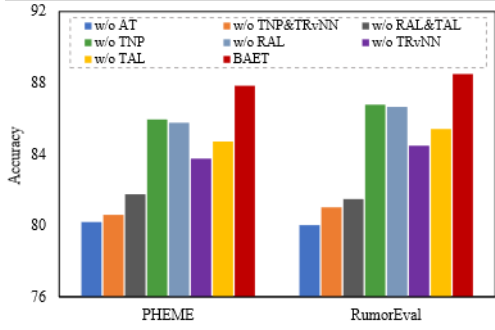


Fig. 4. The post-based ablation analysis in terms of accuracy.

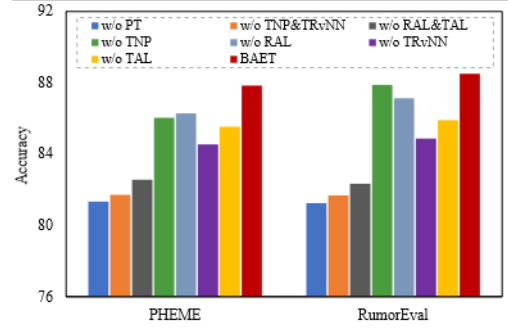


Fig. 5. The author-based ablation analysis in terms of accuracy.

We adopt four metrics *accuracy*, *precision*, *recall*, and *F1-score* to evaluate all comparative rumor detection models. We conducted the experiments using *five-fold cross-validation* and report the average evaluation results. (4) *Platforms*: All experiments were implemented in PyTorch and were conducted on two NVIDIA 1080 12G GPUs.

### B. Overall Experimental Performance

In Table IV, we present the performance of different baseline models in comparison with that our model based on the following four evaluative metrics: *accuracy*, *precision*, *recall* and *F1-score*. The best results are highlighted in bold, and the best baseline results are underlined. From the table, we can make the following observations.

(1) DTR, CNN, and SVM are rumors detection baselines based on handcrafted features. SVM is able to achieve superior results among methods based on handcrafted features. However, we attribute these results to the fact that SVM integrates temporal with textual information, and in doing so simulates information transmission to a certain extent, which increases the relevance between texts.

(2) TE aims to mine rumor text features more thoroughly using social context. However, the table shows that the accuracy of the TE model is significantly lower (nearly 7%) than that of the SVM model, indicating that multi-type feature fusion and information transmission topology may still be considered in order to achieve better performance.

(4) The proposed RvNN architecture was shown to provide a more convincing propagation tree structure for rumor detection. In this sense, RvNNs are able to better combine rumor propagation features and hence improve classification performance. On the other hand, the DTCA architecture was able to achieve good performance by relying on a decision tree mechanism to detect rumors via the establishment of evidence scores while ignoring propagation information. RvNN-GA, with its unique attention mechanism, also achieves significant performance improvements compared with RvNN. However, these two models (RvNN, RvNN-GA) model the propagation topology solely on text contents, ignoring the characteristics of author groups involved in rumor spreading. Meanwhile, the exploration of interactive features between texts is insufficient.

(5) The proposed propagation tree structure broadens the scope of rumor detection research, and researchers focused on modeling the spread of the topological structure in more ways.

BiGCN and EBGCN model the depth and breadth of propagation using graph convolution networks, which has yielded desirable results. Meanwhile, the EBGCN model proposes to investigate the uncertainty of propagation relations, which improves the model's interpretability.

(6) The baselines RGNN and FGCN also achieve desirable performance (the best two baselines). This is reasonable since RGNN and FGCN introduce GNN to capture the post text, author information, and propagation information. However, different from RGNN and FGCN, BAET introducing bipartite adhoc event trees in terms of both posts and authors achieve better performance. The results demonstrate the effectiveness of introducing the author tree and modeling more comprehensive authors' information for rumor detection.

(7) On these datasets, we can see that our proposed BAET outperforms other benchmarks, proving that author information is effective for judging the veracity of the corresponding posts "Birds of a feather propagate together" and our model structure is valid. Simultaneously, it is demonstrated that modeling propagation structure from the perspectives of texts and authors can mine the features gathered through propagation relations, and novel interactive learning modules are introduced to generate a more comprehensive representation of the claim to be detected.

### C. Ablation Study

To verify the contribution of each component of BAET, we perform ablation studies on PHEME and RumorEval, respectively, by discarding certain important components of BAET. Note that our BAET consists of two parallel components for the post tree and the author tree, respectively. Consequently, we perform the ablation study on the post-based and author-based components separately, denoted as post-based ablation and author-based ablation, respectively.

Both post-based and author-based components consist of two modules for node-level representation (including two successive sub-modules TNP and RAL) and structural-level representation (including two successive sub-modules TRvNN and TAL), which are necessary to learn to represent the bipartite adhoc event trees for detecting rumors. To ensure effective ablation studies, all variants of BAET keep the two-level structure, that is at least one sub-module at either node-level or structure-level. Specifically, we compare BAET with

its seven variants in the post-based ablation study and author-based ablation study respectively. We illustrate the variants in the post-based (author-based) ablation study as follows:

- **w/o AT(PT)**: We discard the author (post) tree from BAET, namely using  $\mathbf{h}_{post}$  ( $\mathbf{h}_{author}$ ) only for detection.
- **w/o TNP&TRvNN**: We remove TNP and TRvNN from the node-level representation and structural-level representation modules of BAET, respectively. Accordingly, only the two attention sub-modules, i.e., RAL and TAL, are maintained to learn node/structural representations.
- **w/o RAL&TAL**: We remove the two attention sub-modules, i.e., RAL and TAL, from the node-level representation and structural-level representation modules of BAET, respectively, and feed the node embeddings from TNP to the tree-structural TRvNN to learn to detect.
- **w/o TNP**: We skip the sub-module TNP from the node-level representation module of BAET and feed the raw word embedding embeddings (author feature embeddings) into RAL to the node representations.
- **w/o RAL**: We skip the sub-module RAL from the node-level representation module of BAET and feed the node embeddings from TNP directly into the structural-level representation module to detect.
- **w/o TRvNN**: We remove TRvNN from the structural-level representation module of BAET and feed the node representations into TAL to learn structural representations for detection.
- **w/o TAL**: We remove TAL from the structural-level representation module of BAET and output the last hidden state from TRvNN for detection.

We report the results of the post-based and author-based ablation studies in terms of accuracy in Fig. 4 and Fig. 5, respectively. From the two tables, we observe that the proposed BAET outperforms all other variants in both datasets, indicating the contributions of each modules to detection and verifying the reasonable and effectiveness of our model design in BAET. Specifically, we have the following observations.

(1) The accuracy of the variants **w/o AT** and **w/o PT** is inferior to that of BAET, demonstrating that modeling propagation structure using only posts or authors is insufficient. The results verify that the bipartite adhoc event trees comprehensively summarize the rumor propagation characteristics and that both post texts and author features in circulation contribute to modeling rumor propagation.

(2) The variants **w/o TNP&TRvNN** and **w/o RAL&TAL** fall behind BAET, indicating that TNP&TRvNN and RAL&TAL complement each other and contribute to modeling the propagation information in both the post tree and author tree. The attention modules (RAL&TAL) are helpful to discover post (author) correlations in circulation and learn more informative representations, while TNP and TRvNN involve post/author features and reveal/embed the topological structure, respectively. In addition, the variant **w/o TNP&TRvNN** performs worse than **w/o RAL&TAL**, reflecting the significance of the propagation structure for rumor detection.

(3) The performance degradation of the remaining four variants compared with BAET, reflecting the four sub-modules play different effects and complement each other for rumor de-

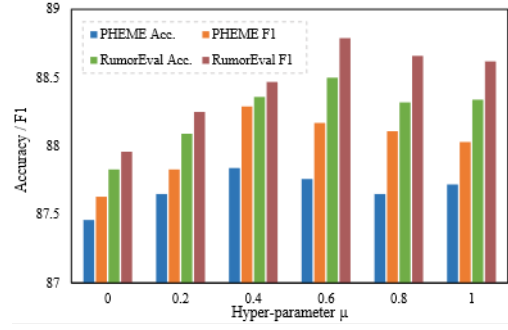


Fig. 6. The accuracy of BAET under different hyper-parameter  $\mu$  on the two datasets.

tection. Specifically, TNP aims to model post text embeddings or author feature embeddings, RAL learns node representations attending to the root node, and TRvNN and TAL capture the topological structure and post (author) correlations, respectively. These designs effectively enhance the representations of rumor propagation from the node and structural levels.

(4) Finally, the performance of the variants in the Fig. 5 is comparable with (slightly worse than) that of the variants in Fig. 4. The results indicate that the author characteristics in the author tree are beneficial for the task of rumor detection. In particular, the users' social attributes and writing habits are involved in learning author node representations. Empirically, these author features facilitate capturing authors' influence (such as reputation and reliability) on rumor propagation and significantly contribute to judging the veracity of rumors.

#### D. Hyper-parameter Effect

(1) First, we investigate the degree and difference to which information from the two interactive modes need to be considered in the *Node-level Representation Module* in Fig. 3 (a) using hyper-parameters  $\mu$ . To complete it, we use Equation 7 and Equation 8. Fig. 6 plots the Accuracy and F1 scores under different  $\mu$  values in *Root-aware Attention Learning* (RAL). From the figure, we can conclude that  $\mu$  of 0.4 on PHEME and 0.6 on RumorEval yield the best performance. According to the dataset introductions in Table III, we speculate that because the PHEME dataset contains a greater number of responsive posts, more propagation information can be captured, and thus the dependency degree of claim information from the claim is slightly lower than that of RumorEval. This demonstrates that the effects of responsive information and claim information on rumor detection results are complementary, and our model also effectively handles the fusion of the two types of them. Furthermore, when ignoring the root-aware attention layer, i.e.,  $\mu = 0.0$ , the performance of BAET gets worse, indicating that ignoring the correlations between the claim node and its successors yields a poor performance of BAET. The results verify the effectiveness and necessity of our proposed root-aware attention layer. Moreover, as  $\mu$  increases, the performance initially improves, while this does not mean that a bigger value of  $\mu$  is always better. If  $\mu$  exceeds a certain value, the model performance exhibits a downward trend, indicating

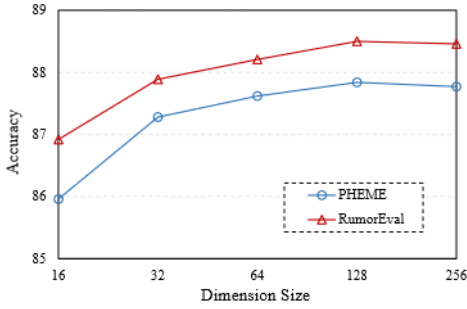


Fig. 7. The accuracy of BAET under different dimension sizes  $d$ .

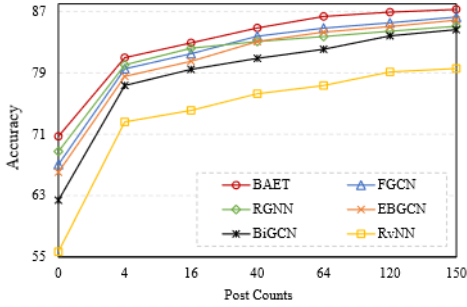


Fig. 8. The accuracy under different responsive post counts on the PHEME.

that the correlation information obtained by two correlation modes may conflict with one another.

(2) Moreover, Fig. 7 shows the performance of our model under different dimension size  $d$ . As shown, our model performance improves as dimension size  $d$  increases. We hypothesize that increasing the embedding and hidden unit dimensions will result in more multidimensional and comprehensive representations of both post texts and author features and that these representations will provide richer information for model training. At the same time, the multi-hierarchical representation learning model that we developed is capable of capturing more interactive information between dimensions. Because our model outperforms on both datasets when  $d = 128$ , we choose 128 as the dimension of embedding and hidden unit. When the dimension size, on the other hand, continues to grow, the model performance gradually improves or worsens. We believe that indefinitely increasing the representation dimension is ineffective. With the increasing dimensions, the model relies on more interactive dimension calculation information, which may increase the possibility of noise information and thus affect the model's detection effect.

#### E. Post Count Effect

To further investigate the importance of the propagation process for rumor detection, we evaluate BAET with five propagation-based baselines (i.e., FGNCN, RGNN, EBGCN, BiGCN, and RvNN) under the different number of retweets or forwarding (i.e., responsive post counts) to which a claim post. We report the accuracy of the comparative models on the PHEME dataset in Fig. 8. As the number of posts increases from 0 to 150, our proposed BAET still outperforms all the baselines, indicating the superiority of BAET even for early

rumor detection (i.e., few posts). When the number of posts is low, all models perform poorly, which is attributed to a lack of relevant information. The veracity of a claim cannot be determined immediately after it is sent out. During the propagation process, responding posts may express approval or disapproval of the claim, as well as provide some relevant evidence in the reply. The pieces of information are gathered through the propagation structure, which provides a solid foundation for identifying the rumor.

As a result, as the number of posts grows, the performance of all models improves. Because multi-perspective modeling of rumor propagation and mining interactive information are taken into account, our model BAET outperforms all others in rumor detection. Second, RvNN proposed an investigation into the rumor detection propagation structure, but its detection capability is inferior to that of other models. Furthermore, while EBGCN and BiGCN have comparable performance, EBGCN has a better detection effect and the BiGCN model has a more consistent performance increase. We hypothesize that the uncertainty of the propagation relation (edges in the tree) considered in the EBGCN model has an auxiliary effect on rumor detection, but because the complexity of the propagation relation increases with the number of posts, the EBGCN model's performance fluctuates slightly.

#### F. Visualization Study

To get a deep insight into the importance of authors who participate in rumors and non-rumors in the author propagation tree, we visualize the attention weights (*Tree-aware Attention Learning* of claims randomly sampled from the PHEME datasets. The result is shown in Fig. 9 with the author propagation tree structure of the first *True Claim* in the left part. Three observations can be made from Fig. 9:

(1) Most of the authors have high weights, indicating that the author's propagation tree structure facilitates the modeling of claim veracity.

(2) For true claims, authors closer to the root author yield a higher weight, i.e., these authors contain more features similar to the root author and "flock" around him, while authors form small-scale flocks during the propagation of false claims. This observation is confirmed by [51] that most people forward true rumors from a centralized source (claim), while rumors are transmitted by people forwarding to others.

(3) Hence, we can infer that to judge a true claim, we could better check the traits of the early forwarding and closer authors. This is not absolute because the experiment in Section V-E shows that the number of responsive posts is also very important for rumor detection; thus, our model needs to be able to integrate multiple factors.

## VI. CONCLUSION

In this study, we define the rumor propagation elements and use the propagation structure to represent the rumor claim as the special event tree structure. To fully exploit the above info, we propose a new rumor detection method (BAET) with hierarchical representation, which models the ad-hoc event tree into bipartite topological trees, namely (*Author*



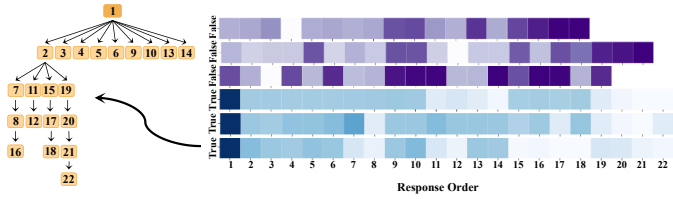


Fig. 9. Visualization of attention weights for author propagation trees of three true and three false claims on PHEME datasets. From left to right is chronological response order; dark colors refer to higher attention weights.

Tree and Post Tree) from two perspectives. And then “flocks” their respective features to obtain a better-integrated learning representation. Meanwhile, in the node-level representation module, we designed word embedding and feature encoder for both post and author separately; also, used root-aware attention learning to capture differentiated correlations between the tree nodes. Similarly, in the structural-level representation module, we adopted a tree-like RvNN network to learn structural dependencies, and tree-aware attention learning was used to mine attention correlations between the tree nodes. Eventually, the evaluation demonstrates that BAET is highly effective, outperforming state-of-the-art baselines.

## REFERENCES

- [1] V. M. Del, B. A., Z. F., P. F., S. A., C. G., S. H. E., and Q. W., “The spreading of misinformation online,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 113, no. 3, 2016.
- [2] Zhiying, Wang, Hongli, Zhao, Huifang, and Nie, “Bibliometric analysis of rumor propagation research through web of science from 1989 to 2019,” *Journal of Statistical Physics*, vol. 178, no. 2, pp. 532–551, 2020.
- [3] D. Lazer, M. A. Baum, Y. Benkler, A. J. Berinsky, K. M. Greenhill, F. Menczer, M. J. Metzger, B. Nyhan, G. Pennycook, and D. Rothschild, “The science of fake news,” *Science*, vol. 359, no. 6380, pp. 1094–1096, 2018.
- [4] L. Cao, Q. Liu, and W. Hou, “COVID-19 modeling: A review,” *CoRR*, vol. abs/2104.12556, 2021.
- [5] A. Zubiaga, A. Aker, K. Bontcheva, M. Liakata, and R. Procter, “Detection and resolution of rumours in social media: A survey,” *ACM Comput. Surv.*, vol. 51, no. 2, pp. 32:1–32:36, 2018.
- [6] D. A. Vega-Oliveros, L. Zhao, A. Rocha, and L. Berton, “Link prediction based on stochastic information diffusion,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 8, pp. 3522–3532, 2022.
- [7] C. Li, H. Peng, J. Li, L. Sun, L. Lyu, L. Wang, P. S. Yu, and L. He, “Joint stance and rumor detection in hierarchical heterogeneous graph,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 6, pp. 2530–2542, 2022.
- [8] C. Castillo, M. Mendoza, and B. Poblete, “Information credibility on twitter,” in *WWW*, 2011, pp. 675–684.
- [9] S. Kwon, M. Cha, K. Jung, W. Chen, and Y. Wang, “Prominent features of rumor propagation in online social media,” in *ICDM*, 2013, pp. 1103–1108.
- [10] M. Lukasik, K. Bontcheva, T. Cohn, A. Zubiaga, M. Liakata, and R. Procter, “Gaussian processes for rumour stance classification in social media,” *ACM Trans. Inf. Syst.*, vol. 37, no. 2, pp. 20:1–20:24, 2019.
- [11] Z. Tan, J. Chen, Q. Kang, M. Zhou, A. Abusorrah, and K. Sedraoui, “Dynamic embedding projection-gated convolutional neural networks for text classification,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 3, pp. 973–982, 2022.
- [12] T. Bian, X. Xiao, T. Xu, P. Zhao, W. Huang, Y. Rong, and J. Huang, “Rumor detection on social media with bi-directional graph convolutional networks,” in *AAAI*, 2020, pp. 549–556.
- [13] A. Lao, C. Shi, and Y. Yang, “Rumor detection with field of linear and non-linear propagation,” in *WWW*, 2021, pp. 3178–3187.
- [14] J. Ma, W. Gao, P. Mitra, S. Kwon, B. J. Jansen, K. Wong, and M. Cha, “Detecting rumors from microblogs with recurrent neural networks,” in *IJCAI*, 2016, pp. 3818–3824.
- [15] T. Chen, X. Li, H. Yin, and J. Zhang, “Call attention to rumors: Deep attention based recurrent neural networks for early rumor detection,” in *PAKDD (Workshops)*, ser. Lecture Notes in Computer Science, vol. 11154, 2018, pp. 40–52.
- [16] K. Wu, S. Yang, and K. Q. Zhu, “False rumors detection on sina weibo by propagation structures,” in *ICDE*, 2015, pp. 651–662.
- [17] J. Ma, W. Gao, and K. Wong, “Detect rumors in microblog posts using propagation structure via kernel learning,” in *ACL (1)*, 2017, pp. 708–717.
- [18] Jing Ma, W. Gao, and K. Wong, “Rumor detection on twitter with tree-structured recursive neural networks,” in *ACL (1)*, 2018, pp. 1980–1989.
- [19] T. Bian, X. Xiao, T. Xu, P. Zhao, W. Huang, Y. Rong, and J. Huang, “Rumor detection on social media with bi-directional graph convolutional networks,” in *AAAI*, 2020, pp. 549–556.
- [20] L. Wei, D. Hu, W. Zhou, Z. Yue, and S. Hu, “Towards propagation uncertainty: Edge-enhanced bayesian graph convolutional networks for rumor detection,” in *ACL/IJCNLP (1)*, 2021, pp. 3845–3854.
- [21] L. Wu, Y. Rao, Y. Zhao, H. Liang, and A. Nazir, “DTCA: decision tree-based co-attention networks for explainable claim verification,” in *ACL*, 2020, pp. 1024–1035.
- [22] Y. Lu and C. Li, “GCAN: graph-aware co-attention networks for explainable fake news detection on social media,” in *ACL*, 2020, pp. 505–514.
- [23] N. Difonzo and P. Bordia, “Rumor, gossip and urban legends,” *Diogenes*, vol. 54, no. 1, pp. 19–35, 2007.
- [24] M. E. Jaeger, S. Anthony, and R. L. Rosnow, “Who hears what from whom and with what effect: A study of rumor,” *Personality & Social Psychology Bulletin*, vol. 6, no. 3, pp. 473–478, 1980.
- [25] R. L. Rosnow and E. K. Foster, “Rumor and gossip research,” *Psychological Science Agenda*, vol. 19, no. 4, 2005.
- [26] V. Qazvinian, E. Rosengren, D. R. Radev, and Q. Mei, “Rumor has it: Identifying misinformation in microblogs,” in *EMNLP*, 2011, pp. 1589–1599.
- [27] A. Hannak, D. Margolin, B. Keegan, and I. Weber, “Get back! you don’t know me like that: The social mediation of fact checking interventions in twitter conversations,” in *ICWSM*, 2014.
- [28] Z. Zhao, P. Resnick, and Q. Mei, “Enquiring minds: Early detection of rumors in social media from enquiry posts,” in *WWW*, 2015, pp. 1395–1405.
- [29] F. Yang, Y. Liu, X. Yu, and M. Yang, “Automatic detection of rumor on sina weibo,” in *SIGKDD Workshop on Mining Data Semantics*, New York, NY, USA, 2012.
- [30] Y. Chen, Z. Liu, and H. Kao, “IKM at semeval-2017 task 8: Convolutional neural networks for stance detection and rumor verification,” in *SemEval@ACL*, 2017, pp. 465–469.
- [31] J. Ma, W. Gao, Z. Wei, Y. Lu, and K. Wong, “Detect rumors using time series of social context information on microblogging websites,” in *CIKM*, 2015, pp. 1751–1754.
- [32] B. Rath, W. Gao, J. Ma, and J. Srivastava, “Utilizing computational trust to identify rumor spreaders on twitter,” *Soc. Netw. Anal. Min.*, vol. 8, no. 1, pp. 64:1–64:16, 2018.
- [33] F. Yu, Q. Liu, S. Wu, L. Wang, and T. Tan, “Attention-based convolutional approach for misinformation identification from massive and noisy microblog posts,” *Comput. Secur.*, vol. 83, pp. 106–121, 2019.
- [34] Q. Zhang, L. Cao, C. Shi, and Z. Niu, “Neural time-aware sequential recommendation by jointly modeling preference dynamics and explicit feature couplings,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 10, pp. 5125–5137, 2022.
- [35] L. M. S. Khoo, H. L. Chieu, Z. Qian, and J. Jiang, “Interpretable rumor detection in microblogs by attending to user interactions,” in *AAAI*, 2020, pp. 8783–8790.
- [36] K. Shu, X. Zhou, S. Wang, R. Zafarani, and H. Liu, “The role of user profiles for fake news detection,” in *ASONAM*, 2019, pp. 436–439.
- [37] N. Ruchansky, S. Seo, and Y. Liu, “CSI: A hybrid deep model for fake news detection,” in *CIKM*, 2017, pp. 797–806.
- [38] K. Shu, S. Wang, and H. Liu, “Beyond news contents: The role of social context for fake news detection,” in *WSDM*, 2019, pp. 312–320.
- [39] X. Zhou and R. Zafarani, “A survey of fake news: Fundamental theories, detection methods, and opportunities,” *ACM Comput. Surv.*, vol. 53, no. 5, pp. 109:1–109:40, 2020.
- [40] S. Wang and T. Terano, “Detecting rumor patterns in streaming social media,” in *IEEE BigData*, 2015, pp. 2709–2715.
- [41] R. Socher, C. C. Lin, A. Y. Ng, and C. D. Manning, “Parsing natural scenes and natural language with recursive neural networks,” in *ICML*, 2011, pp. 129–136.

- [42] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. Manning, A. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," *EMNLP*, vol. 1631, pp. 1631–1642, 01 2013.
- [43] J. Ma, W. Gao, S. R. Joty, and K. Wong, "An attention-based rumor detection model with tree-structured recursive neural networks," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 4, pp. 42:1–42:28, 2020.
- [44] H. Lin, J. Ma, M. Cheng, Z. Yang, L. Chen, and G. Chen, "Rumor detection on twitter with claim-guided hierarchical graph attention networks," in *EMNLP (1)*, 2021, pp. 10035–10047.
- [45] B. Liu, X. Sun, Q. Meng, X. Yang, Y. Lee, J. Cao, J. Luo, and R. K.-W. Lee, "Nowhere to hide: Online rumor detection based on retweeting graph neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–12, 2022.
- [46] L. Wei, D. Hu, W. Zhou, X. Wang, and S. Hu, "Modeling the uncertainty of information propagation for rumor detection: A neuro-fuzzy approach," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–12, 2022.
- [47] A. Zubiaga, M. Liakata, R. Procter, G. W. S. Hoi, and P. Tolmie, "Analysing how people orient to and spread rumours in social media by looking at conversational threads," *PLoS ONE*, vol. 11, no. 3, 2016.
- [48] L. Derczynski, K. Bontcheva, M. Liakata, R. Procter, G. W. S. Hoi, and A. Zubiaga, "Semeval-2017 task 8: Rumoureal: Determining rumour veracity and support for rumours," in *SemEval@ACL*, 2017, pp. 69–76.
- [49] G. B. Guacho, S. Abdali, N. Shah, and E. E. Papalexakis, "Semi-supervised content-based detection of misinformation via tensor embeddings," in *ASONAM*, 2018, pp. 322–325.
- [50] J. C. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, 2011.
- [51] Z. Zhao, J. Zhao, Y. Sano, O. Levy, H. Takayasu, M. Takayasu, D. Li, J. Wu, and S. Havlin, "Fake news propagates differently from real news even at early stages of spreading," *EPJ Data Sci.*, vol. 9, no. 1, p. 7, 2020.

**Qi Zhang** received his Ph.D. degrees from Beijing Institute of Technology China under the dual Ph.D. program of Beijing Institute of Technology and University of Technology Sydney Australia. He is currently currently a research fellow in Tongji University. He has published a number of papers in premier conferences and journals, including AAAI, IJCAI, SIGIR, TheWebConf, TKDE, and TNNLS. His primary research interests focus on collaborative filtering, sequential recommendation, learning to hash, and MTS analysis.

**Yayi Yang** is currently studying her master degree in Beijing Institute of Technology China in Computer Science. Her research interests focus on rumor detection, multi-modal fake news detection and sentiment analysis.

**Chongyang Shi** is a currently associate professor in School of Computer Science, Beijing Institute of Technology. He received his PhD degree from BIT in 2010 in Computer Science. Dr. Shi's research areas focus on information retrieval, knowledge engineering, personalized service, sentiment analysis etc. He serves as an editorial board member for several international journals, and has published more than 20 papers in international journals and conferences.

**An Lao** is studying her master's degree from Beijing Institute of Technology. Her primary research interests include rumor detection, sentiment analysis and multi-modal fusion. She will pursue her PhD at Beijing Institute of Technology.

**Liang Hu** received dual Ph.D. degrees from Shanghai Jiao Tong University, China and University of Technology Sydney, Australia. He is currently a distinguished research fellow in Tongji University and chief AI scientist in DeepBlue Academy of Sciences. His research interests include recommender systems, machine learning, data science and general intelligence.

**Shoujin Wang** received a PhD in Data Science from the University of Technology Sydney (UTS) in 2019. He is currently a Lecturer in Data Science at UTS, Australia. His research interests include data mining, machine learning, recommender systems, fake news mitigation. He has published a number of papers in premier conferences and journals, including TheWebConf, AAAI, IJCAI, ECML-PKDD, ACM CSUR. He is a receipt of some prestigious awards, including 2022 DSAA Next-generation Data Scientist Award and 2022 Club Melbourne Fellowship Award.

**Usman Naseem** is a Ph.D. student in the School of Computer Science, The University of Sydney, Australia. He received his master's degree in Analytics (research) from the School of Computer Science, University of Technology Sydney, Australia, in 2020. His primary research is in the intersection of machine learning and natural language processing for social media analytics and biomedical/health informatics.