Catch me if you can: a participant-level rumor detection framework via fine-grained user representation learning

Chen, X.; Zhouf, F.; Zhang, F.L.; Bonsangue, M.M.

Information Processing and Management 58 (2021) 102678

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/ipm

Catch me if you can: A participant-level rumor detection framework via fine-grained user representation learning

Xueqin Chen [a,b], Fan Zhou [a,*], Fengli Zhang [a], Marcello Bonsangue [b]

[a] University of Electronic Science and Technology of China, China
[b] Leiden University, Netherlands

ARTICLE INFO

Keywords:
Rumor detection
Participant-level
User influence
Susceptibility
Temporal

ABSTRACT

Researchers have exerted tremendous effort in designing ways to detect and identify rumors automatically. Traditional approaches focus on feature engineering. They require lots of human actions and are difficult to generalize. Deep learning solutions come to help. However, they usually fail to capture the underlying structure of the rumor propagation and the influence of all participants involved in the spreading chain. In this study, we propose a novel participant-level rumor detection framework. It explicitly models and integrates various fine-grained user representations (i.e., user influence, susceptibility, and temporal information) of all participants from the propagation threads via deep representation learning. Experiments conducted on real-world datasets demonstrate a significant accuracy improvement of our approach. Theoretically, we contribute to the effective usage of data science and analytics for social information diffusion design, particularly rumor detection. Practically, our results can be used to improve the quality of rumor detection services for social platforms.

1. Introduction

The rapid development of Internet technology has democratized the exchange of information. Online social platforms, such as Twitter, Yelp, Reddit, etc., have emerged and gradually become the primary source of information to guide individuals' everyday decisions. According to Pew Research,[1] as of 2017 approximately 88% of American adults have either free or paid Internet access at home, and about 81% obtain news from online platforms (e.g., news websites/apps, social media, or both). While Internet technology and social media facilitate free information creation and sharing, the proliferation of fake news, rumors, and false information has had strong and negative societal and economic consequences. The explosive spreading of false information poses a threat to the credibility of legitimate online platforms and resources and has a severe negative impact on both individuals and society (Shu et al., 2017a), with the potential consequences to destabilize nations, affect the fairness of competition (Allcott & Gentzkow, 2017), and shock the stock market (DiFonzo & Bordia, 1997). For example, in the global effort to contain the COVID-19 pandemic, misinformation abounds and flourishes on the Internet, and people have been led to believe that COVID-19 can be cured by ingesting fish tank cleaning products or that 5G networks generate radiation that triggers the virus. Such misinformation not only causes panic among citizens but could potentially undercut collective efforts to control the pandemic.

Detecting fake news on online social platforms as early as possible is therefore necessary, urgent, and socially beneficial. To develop an effective rumor detection that can identify different types of rumors with high accuracy. This is necessary because, for

* Corresponding author.
E-mail addresses: nedchen0728@gmail.com (X. Chen), fan.zhou@uestc.edu.cn (F. Zhou), fzhang@uestc.edu.cn (F. Zhang), m.m.bonsangue@liacs.leidenuniv.nl (M. Bonsangue).

[1] https://www.journalism.org/2016/07/07/pathways-to-news/.

https://doi.org/10.1016/j.ipm.2021.102678
Received 20 January 2021; Received in revised form 8 April 2021; Accepted 25 June 2021
Available online 13 July 2021

example, poorly classifying non-rumor news as rumor news and blocking it from a social media platform may backfire, as such an inappropriate action undermines the fundamental information-sharing purpose of the social media platform (Shu et al., 2017b; Zhang & Ghorbani, 2020; Zhou & Zafarani, 2020).

To develop our approach, we considered the life cycle of real and fake news on social media, which plays a crucial role in information diffusion. When the news is produced by the content creator, it starts its journey on the social media platform. Once people are exposed to the news, they become the content consumers. According to the confirmation bias theory, people tend to favor, interpret and share information in a way that confirms or strengthens their prior beliefs or ideologies. As a result, if a news item confirms the consumers' prior ideology, they may share it within their social networks in the role of content distributor. Since fake news is intentionally written to mislead readers into believing and propagating false information (e.g., 5G networks trigger COVID-19), it is plausible that fake news is more easily distributed among its believers than real news, which is neutral in its beliefs and ideology. This idea is supported by prior studies, which have noted that false information tends to spread significantly faster, further, deeper, and more broadly than real information (Vosoughi et al., 2018). Therefore, considering all participants, including content creators and content distributors, in the news diffusion chain may improve the overall rumor detection performance of our approach.

Another design consideration is the lack of effective methods to represent all participants in the news diffusion chain. Prior predictive studies have simply used aggregated statistics, such as the total number of content distributors (retweets) and the average time of information distribution, to quantify the diffusion process. This inevitably results in information loss and suboptimal performance. Other examples of such aggregated statistics include network-level attributes (e.g., density) to represent diffusion networks, the final hidden representation from recurrent neural networks to model the temporal spreading sequence, and overall descriptive statistics of user characteristics (e.g., mean user tenure) to describe users in the diffusion (Castillo et al., 2011). While such data may be helpful in modeling, they are not quite specific enough to provide a clear picture of by whom, when, why, and how news is diffused. Therefore, the key question motivating our study is how to design an effective predictive method that represents fine-grained all-participant patterns throughout the whole diffusion process, including social patterns (user social homophily), diffusion patterns (user influence and susceptibility), temporal patterns (how fast the news is propagated in the social network), all-participant profile patterns.

In our work, we propose a novel framework based on deep representation learning for rumor detection, named PLRD (*P*articipant-*L*evel *R*umor *D*etection). In view of theories on propagation and social influence, PLRD incorporates multi-scale features of all users[2] enrolled in the diffusion process to predict a given post's credibility (e.g., classify it as rumor or non-rumor). Specifically, PLRD first employs sparse matrix factorization to embed the global graph (i.e., user-interaction graph constructed on all propagation threads), which can efficiently learn the social homophily for users. Then, it uses a multi-hop graph convolutional layer, and a bi-directional GRU to learn fine-grained user representations (i.e., the user influence, user susceptibility, and user temporal). To understand the different importance of users' multi-scale representations, a feature-level attention layer was designed to explain which types of features are essential in rumor propagation. Moreover, to capture the uncertainty in learned features, PLRD introduces a variational autoencoder. Finally, PLRD employs a user-level attention layer to allocate different importance to users and aggregate them to form a unique rumor representation. The rumor prediction is generated based on the learned unique representation.

Our design science work on rumor detection makes two main contributions to the literature in this field. First, our design is rooted in social influence and propagation theory, from which we derive various constructs in our model. To the best of our knowledge, our approach is among the first to detect rumors at a very fine-grained participant level. It is very different from prior works that also combine information from various sources (e.g., reply networks, diffusion sequence, and attributes of spreading users) but previous studies still heavily depend on the text features, our approach comprehensively exploits user-profiles and propagation threads and shows a strong ability to detect rumors without using any text information. Second, we make a methodological contribution by proposing an approach to learn fine-grained user representation via deep representation learning that effectively captures all participant information in a diffusion chain. This information includes user influence, user susceptibility, and user temporal. Experimental results using real-world datasets confirm the effectiveness of our approach over prior rumor detection methods. Our approach has direct implications for social media platforms that are vulnerable to rumor spreading since it can be deployed to identify original users who initiate rumors and those who spread rumors. Overall, the proposed rumor-detection model can help improve the user experience and benefit society by helping individuals obtain healthy and genuine information.

The rest of the paper is organized as follows. Section 2 reviews related work, then Section 3 introduces the theoretical foundations of our study, as well as the data and some preliminary background. Section 4 presents the major aspects of PLRD methodology in detail. Experiment evaluations quantifying the benefit of our approach are described in Section 5. Section 6 concludes the paper and outlines directions for future work.

2. Related work

We review prior predictive studies on rumor detection, which generally fall into two categories: the feature engineering-based approaches and the deep learning-based models.

[2] In our work, the user influence, user susceptibility and user temporal are collectively referred to as multi-scale information of users.

2.1. Feature engineering in rumor detection

In the area of rumor detection, handcrafted feature engineering-based approaches are used to extract various relevant features from raw data. The features are typically fall into two major categories: (1) content features extracted from the text (e.g., characters, words, sentences, and documents) and visual elements (e.g., images and videos) (Castillo et al., 2011; Gupta et al., 2012; Jin et al., 2017; Kwon et al., 2013; Wu et al., 2015; Zhao et al., 2015); and (2) social context features extracted from the user behavior and the diffusion network, which reflect the relationships among users and describe the diffusion process of a rumor, such as user demographics (Shu et al., 2018, 2019b), propagation activities (Ma et al., 2017; Wu et al., 2015), and diffusion temporality (Ma et al., 2015). Both content features and social context features are fed into discriminative machine learning algorithms to perform rumor detection.

Since rumors are intentionally written to mislead readers into believing and propagating false information, their textual content often tends to have certain patterns compared to non-rumors. One study (Zhao et al., 2015) describes two types of language patterns (inquiry and correction patterns) in rumors, then detects the patterns of rumor messages through supervised feature selection. Similarly, Wu et al. (2015) extract a set of topic features to summarize semantics and train a Latent Dirichlet Allocation (LDA) model for detecting rumors on Weibo. Moving towards a more comprehensive understanding of text on social media, other studies have derived non-general textual features, such as emotions Castillo et al. (2011), from social media platforms.

Social context features are derived from the social connection characteristics of social media users. Rumors are usually created by a few users but spread by many. User profiles can be used to measure the user's characteristics and credibility (Castillo et al., 2011). Kwon et al. (2013) extended this work and further proposed 15 structural features extracted from the diffusion network as well as the user friendship network. Recently, Shu et al. (2019b) further study the effectiveness of user profiles in rumor detection.

We note that the above approaches' performance heavily depends on the handcrafted features, for which there is no standard and systematic design protocol. In fact, the conclusions of existing works usually contradict each other, primarily due to the different types of datasets used to inform feature design. In our work, instead of manually creating features, we leverage deep learning to represent the fine-grained diffusion process.

2.2. Deep learning techniques in rumor detection

Due to the recent success of deep learning in different fields, such as NLP (Natural Language Processing) and CV (Computer Vision), much attention has been paid to developing a deep neural network-based method for rumor detection. Various studies have already shown that such methods offer a performance improvement due to their enhanced ability to extract relevant features and represent data. The first study to apply recurrent neural networks (RNN) to model rumors as varied length time series aimed to learn both temporal and textual features from raw data and thus detect rumors (Ma et al., 2016). Other researchers built a tree-structured RNN to catch hidden representations from both propagation structures and text content. For example, Yu et al. proposed a convolutional neural network-based approach to handle the problem occurring in RNN-based methods. The motivation is that RNN is not suitable to perform early rumor detection with limited inputting data, and it has a bias towards the latest elements of the input sequence (Yu et al., 2017). Simultaneously, plenty of methods use a combination of CNN and RNN. For example, Yang et al. built a time series classifier with both RNN and CNN to predict whether a given news story is fake at an early stage, which takes common users' characteristics and propagation paths into consideration (Liu & Wu, 2018).

A few recent studies explore other deep neural networks for rumor detection. For instance, Bian et al. proposed a GCN-based model to learn global structural relationships of rumor dispersion (Bian et al., 2020). Li et al. also employed a graph-based model to explore the homogeneity and conversation structure in rumor spreading (Li et al., 2020). Yang et al. proposed an adversarial learning-based model, taking into account the camouflages of rumors from an adversarial perspective (Yang et al., 2020). Yu et al. implemented a fusion network with the combination of multilayer perceptron and GCN to fuse and learn rumor representation from a set of static features (Yu et al., 2020). Wu et al. first constructed a decision tree-based evidence model to select high credibility comments as evidence and then designed an attention-based network to capture the false parts of the claims (Wu et al., 2020). Inspired by the multi-task learning scheme, Ma et al. proposed two multi-task architectures based on RNNs for joint training the task of stance classification and rumor detection (Ma et al., 2018a).

While the aforementioned models achieve state-of-the-art performance and demonstrate that fusing different kinds of features can improve rumor detection performance, they still exhibit several drawbacks. Specifically, these models still heavily depend on linguistic-based content features, and they are inefficient in learning the propagation structure and user temporal influence. Furthermore, no existing method has modeled rumor detection at a fine-grained all-participant level, which motivates our present study.

3. Birds of a feather flock together: the perspective of all participants

Rumor detection has long been a subject of interdisciplinary research. Various theories have been proposed and validated. In this section, we discuss several major theories that guide us to derive relevant constructs in our model for better rumor detection.

Table 1
Statistics of the datasets.

Statistic	Science	Twitter15	Twitter16	RumourEval19
# source tweets	16,901	739	404	271
# users	3,603,265	306,402	168,659	4,774
# edges of global graph	3,586,360	336,486	182,493	4,503
# non-rumors	14,442	370	199	167
# rumors	2,459	369	205	104
Max. # retweets	46,895	2,990	999	155
Min. # retweets	5	97	100	3
Avg. # retweets	213	493	479	18
Avg. # time length	749 h	743 h	167 h	37 h

3.1. Theory

Users play major roles in the dissemination of rumors or fake news. A set of user-based and propagation-based theories were developed to study how rumor spreads, how users engage with rumor, and the role users play in rumor creation, propagation, or intervention. For example, in the echo chamber effect, individuals tend to believe information is correct after repeated exposures (Shore et al., 2018). Confirmation bias theory tells us that individuals tend to trust information that confirms their preexisting beliefs or hypotheses, which they perceive to surpass that of others (Nickerson, 1998). People choose to interact with those who share similar opinions and avoid those with whom they profoundly disagree. Both indicate that people may react to and process information differently based on information type (e.g., rumor vs. non-rumor). On the other hand, homophily theory says that individuals in homophilic relationships share common characteristics (beliefs, demographics, etc.) that make communication easier (McPherson et al., 2001). Meanwhile, social identity theory shows that individuals do something primarily because others are doing it and to conform in order to be liked and accepted by others. Such social influence and homophilic atmospheres also exist in and are commonly seen in online social networks (Kelman, 1958).

In sum, all the above considerations suggest that user attributes likely have an impact on rumor detection, such as user influence, susceptibility, and temporal, etc. This supposition is also proven by our data exploration (see below) and computational experiments (see the Evaluation section). We therefore hypothesize that:

Hypothesis 1. *Combining various information at a fine-grained all-participant level in a diffusion chain will improve rumor discovery performance.*

Hypothesis 2. *Deep representation learning-based methods will improve rumor discovery performance compared to using shallow machine learning methods.*

3.2. Data

In our work, we conduct experiments on four standard real-word testbeds: Twitter15, Twitter16, Science and RumourEval19, all of which were collected from Twitter,[3] one of the most popular social media platforms in the U.S. The descriptive statistics of all datasets are shown in Table 1.

Twitter15/16.[4] were released by Ma et al. (2017) The two datasets share the same structure and contain the tweets (events) and retweets from news articles published in 2015 and 2016. For each article, the data contains the first tweet shared on Twitter and a sequence of retweets following this initial post. In Twitter15 and Twitter16, we keep the tweets labeled as "non-rumor" or "false rumor" (relabeled as "rumor" in our work), since the others were beyond our research interest.

Science,[5] is collected and studied by Vosoughi et al. (2018) which includes complete retweet cascades linked to rumors that were verified and published by fact-checking websites. In the original data, each tweet cascade is related to a specific label, i.e., "TRUE", "FALSE" or "MIXED". In our work, we remain the tweets labeled as "TRUE" or "FALSE" (relabeled as "non-rumor" and "rumor" in our work, respectively). To the best of our knowledge, the Science dataset is the most credible dataset among all the existing Twitter-based rumor detection datasets as it overcomes issues of partiality or bias because of the sampling restriction characteristic. In this work, we use the Science dataset to provide model-free evidence to support the **Hypothesis 1**.

RumourEval19,[6] (Gorrell et al., 2019) is an extensive dataset from RumourEval17 (Derczynski et al., 2017) which is augmented with new Twitter test data and Reddit material. In this work, we keep an eye on Twitter data and ignore the Reddit material, and finally obtain 381 Twitter conversation threads. Each thread consists of a claim and a tree of comments, and is related to a specific label, i.e., "true", "false" or "unverified". We filter out unverified tweets and finally get 271 Twitter conversation threads (relabeled as "non-rumor" and "rumor", respectively). Because the SemEval19 contains rich textual information rather than diffusion patterns, in the experiments part, we use SemEval19 to test our model performance when including textual features.

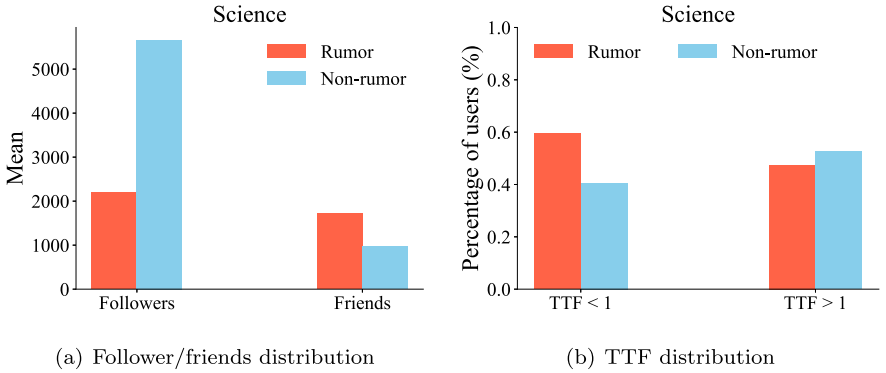
[3] https://twitter.com/.
[4] https://www.dropboxcom/s/46r50ctrfa0ur1o/rumdect.zip.
[5] Researchers interested in gaining access to Science dataset should contact Vosoughi et al. (2018) directly.
[6] https://competitions.codalab.org/competitions/19938.

Table 2
Summary of user characteristics for Twiter15/16 and RumourEval19.

No.	Characteristic	Data Type
1	LENGTH OF USER NAME	Integer
2	USER COUNT CREATED TIME	Integer
3	LENGTH OF DESCRIPTION	Integer
4	FOLLOWERS COUNT	Integer
5	FRIENDS COUNT	Integer
6	STATUSES COUNT	Integer
7	IS VERIFIED	Binary
8	IS GEO ENABLED	Binary

Table 3
Summary of user characteristics for Science.

No.	Characteristic	Data Type
1	USER COUNT CREATED TIME	Integer
2	FOLLOWERS COUNT	Integer
3	FRIENDS COUNT	Integer
4	ENGAGEMENT	Float
5	IS VERIFIED	Binary



(a) Follower/friends distribution (b) TTF distribution

Fig. 1. Social influence/susceptibility analysis on Science.

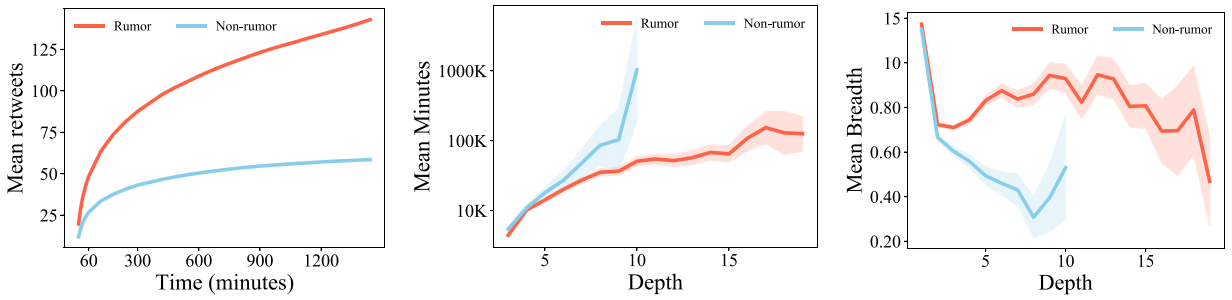
For each tweet, we construct the diffusion graph and global graph from the propagation threads (see Definition 1 and Definition 3 for formal definitions). In Twitter15/16, no user information is provided due to constraints set out in Twitter's terms of service. We crawl all the related user profiles via *Twitter API*[7] based on the provided user IDs. We follow the work of "PPC_RNN+CNN" (Liu & Wu, 2018) and select 8 general user characteristics for experiments, which are summarized in Table 2. As for Science, all data, such as ids, were anonymized, so we directly use the user characteristics provided in this dataset, which are listed in Table 3. Here, the concrete definitions of each characteristic can be found in the supplementary of (Vosoughi et al., 2018). The RumourEval19 provided JSON files for each source tweet and its corresponding replies, and each file contains the complete information of the tweet and the users.

3.3. Model-free evidence

Following from our earlier **Hypothesis 1** that utilizing the information of all users who participated in a diffusion chain might improve rumor detection, we first check for any patterns or differences among involved participants in terms of their overall attributes across the rumors and non-rumors via analyzing the Science dataset.

- **Social influence/susceptibility: followers/friends.** Social influence may affect the speed and the depth of diffusion. On Twitter, a user's social influence can be measured by the size of their social circle in relation to two factors: the number of friends (i.e., users the specific user is following) and the number of followers (i.e., users following the specific user). We calculate the average followers/friends across all participants and find that this average in the rumors group is different from that in the non-rumors group (see Fig. 1(a)). To compute the social influence of all participants, we define a new metric, *TTF*, which combines followers and friends: $TTF = \frac{\#\text{followers}}{\#\text{friends}}$. Users with $TTF < 1$ are less influential but with higher susceptibility,

[7] https://dev.twitter.com/rest/public.



(a) Propagation speed (b) Depth-time (c) Depth-breadth

Fig. 2. Structural and temporal analysis on Science.

Table 4
List of Notations.

Symbol	Description
M	A set of tweets/posts.
m_i	A specific tweet/post.
$\mathcal{G}$	Global graph.
U, E, N	User set, edge set and the number of nodes of global graph.
G_i, G_i^T	Diffusion graph and inverse diffusion graph of m_i.
$\mathbf{U}_i$	User characteristic matrix of tweet m_i.
$U_i, E_i, \lvert U_i \rvert$	User set, edge set and the number of users of tweet m_i.
T_o, t_*	The observation window, time-stamp for each user in tweet m_i.
$\mathbf{E}^{\mathcal{G}}, \mathbf{e}_*^{\mathcal{G}}$	The user social homophily embedding matrix and social homophily vector for user u_*.
$\mathbf{T}_i$	Time-embedding of m_i.
$\mathbf{H}_i^{inf}, \mathbf{H}_i^{sus}, \mathbf{H}_i^{temp}$	The representations of the user influence, susceptibility and temporal respectively.
$\tilde{\mathbf{U}}_i, \mathbf{U}_i^z, \mathbf{U}_i^F$	The representation after feature-level attention, VAE and concatenate operation of m_i.
$\mathbf{R}_i$	The final representation of tweet m_i.
$\hat{\mathbf{Y}}/\hat{\mathbf{y}}_*, \mathbf{Y}/\mathbf{y}_*$	The predicted label and the ground truth.

since they have fewer followers than friends, and extreme cases are fake users. In contrast, users with $TTF > 1$ are more influential, e.g., celebrity accounts. Fig. 1(b) shows that a higher percentage of users with $TTF < 1$ are involved in rumors, while more influential users participate in non-rumors.

- **Structural and temporal impact of diffusion.** Many prior studies have demonstrated that falsehood diffuses significantly farther, faster, deeper, and more broadly than the truth in all categories of information (Vosoughi et al., 2018). This being the case, we expect to see that the spreading pattern, in terms of network structures and the temporal sequence of retweeting, should vary based on rumor type. To this goal, we analyze the propagation speed, depth-time distribution and depth-breadth distribution, respectively, which are shown in . In the diffusion graph, the depth of a node is the number of hops from the node to the source node, and the breadth at a specific depth of a graph is the total number of nodes at this depth level. From Fig. 2(a), we can find that at the same time-scale, rumors can infect more users than non-rumors, which demonstrates that rumors spreading faster than non-rumors. In Fig. 2(b), we measured the average time (in minutes) rumor or non-rumor tweets took to reach different depths, where we observe that (1) rumors can reach deeper users than non-rumors, and (2) rumors require less time to reach the same depth with non-rumors. In Fig. 2(c), we plot the average breadth at every depth for rumors and non-rumors, which indicates that rumors spread broader than non-rumors, especially at a deeper level. These observations substantiate our belief that incorporating such structural and temporal information into the model may improve rumor detection performance.

4. Methodology

In this section, we first present a preliminary overview of rumor detection in our context and describe the overall framework of the proposed PLRD method, followed by details of each component in the model. As discussed above, two challenges need to be addressed when designing an effective rumor detection system: (1) how to incorporate fine-grained all-participant information in one model – not simply aggregating or concatenating – to capture rumor spreading patterns; and (2) how to effectively learn latent representations of users' propagation activities in a diffusion chain to capture fine-grained user representations. To answer these two questions, we first formally define our problem and describe its context.

4.1. Preliminaries and problem statement

In this section, we give the necessary background and formally define the rumor detection problem. We list the main mathematical notations used throughout the paper in Table 4.

In this study, we formalize our rumor detection problem as a supervised binary-class classification task. Suppose the input of the task is from a rumor detection dataset (e.g., Twitter) consisting of a set of posts (e.g., tweets) denoted as $M = \{m_i, i \in [1, |M|]\}$. Each m_i corresponds to its own diffusion process and all participating users, so that m_i can be represented by $\{G_i, \mathbf{U}_i\}$, where G_i and $\mathbf{U}_i$ are the diffusion graph and the user characteristics matrix, respectively. In addition, we construct a global graph $\mathcal{G}$ based on all tweets M, to represent the social homophily among all users. See their formal definitions below.

Definition 1 (*Diffusion Graph G*)**.** A diffusion graph for tweet m_i is denoted as $G_i = \{U_i, E_i\}$, where U_i is the nodes set comprising $|U_i|$ nodes (i.e., all users who are involved in the diffusion process of m_i), $E_i = \{(u_i, u_j, t_j)|u_i, u_j \in U_i\}$ represents a set of edges connecting pairs of users when u_j retweets u_i at time t_j. Note that, in our work, the diffusion graph is a directed acyclic graph.

Definition 2 (*User Characteristics Matrix* $\mathbf{U}_i$)**.** Each user $u_j \in U_i$ is associated with a user vector $\mathbf{u}_j \in \mathbb{R}^{d_{user}}$, which represents all relevant user profile information, such as number of followers, whether the user is verified, etc. We concatenate vectors of all users who retweeted the tweet m_i to form a user characteristics matrix $\mathbf{U}_i$. The concatenation order follows the diffusion time.

Definition 3 (*Global Graph* $\mathcal{G}$)**.** The global graph $\mathcal{G} = \{U, E\}$ is a collection of nodes and edges, which is constructed based on all posts in the dataset. $U = \bigcup_{i=1}^{|M|} U_i$ is a user set contains all users in dataset, and E is the edge set. An edge between u_i and u_j refers to these two users share the same tweet (or discuss the same topic).

Definition 4 (*Rumor Detection*)**.** Given a tweet $m_i = \{G_i, \mathbf{U}_i|T_o\}$ within an observation window T_o (in our work, T_o is the total number of retweets), and the global graph $\mathcal{G}$, the goal of rumor detection is to learn a function $f(\hat{y}_i|m_i, \mathcal{G})$ to classify the source tweet m_i into one of the rumor categories, where the predicted result $\hat{y}_i$ represents either non-rumor or rumor.

4.2. Overall framework of PLRD

In this section, we describe our proposed PLRD rumor detection system. It consists of the following components (see Fig. 3): (a) inputs, including (1) the propagation threads of tweet m_i, (2) global graph constructed on all tweets propagation threads; (b) preprocessing layer, which has: (1) construct diffusion graph and inverse diffusion graph based on propagation threads, (2) construct user characteristic matrix based on user profiles, (3) pre-train retweet time-stamps via positional encoding, and (4) pre-train global graph via randomized truncated singular value decomposition-based sparse matrix factorization; and (c) fine-grained user representation learning layer and rumor detection layer, in which we learn user influence and susceptibility via multi-hop graph convolution layer, model user temporal feature via bi-directional GRU, then aggregate and enhance the learned multi-scale user representations through a feature-level attention layer and a variational autoencoder, finally, after a user-level attention layer, we feed the unique rumor representation into a rumor classifier. Specifically, we use several fully connected feedforward layers and a softmax output layer to generate a rumor prediction. Below, we explain each of the above components in detail.

4.3. Social homophily learning from global graph

As mentioned in Definition 1 and Definition 3, we construct the global graph based on all the retweet threads in the dataset. Our goal is to capture social homophily for all users. The social homophily among users specifies that users with similar interests are more likely to closely connected (Sankar et al., 2020). We assume that the users who discuss the same post in social communities share homophilous relationships in our work. Then, we try to model social homophily in an unsupervised manner, which encourages users with shared social neighborhoods to have similar latent representation.

More specifically, we cast the problem of learning social homophily as the task of graph embedding. The global graph $\mathcal{G}$ always contains tens of thousands of nodes, and to model such a large graph effectively is a tough challenge in the field of graph representation learning (Cao et al., 2015; Zhang et al., 2019). Inspired by the success of sparse matrix factorization (SMF) in large-sized graph representation learning, in our work, we use a randomized tSVD-based SMF to learn social homophily from the global graph. Here, tSVD is truncated singular value decomposition, which can prevent the problem of infeasible computation of factorization for a large-sized matrix (Halko et al., 2011; Tao, 2012). Specifically, given global graph $\mathcal{G}$, we can obtain the adjacency matrix $\mathbf{A}^{\mathcal{G}} \in \mathbb{R}^{N \times N}$ and diagonal degree matrix $\mathbf{D}^{\mathcal{G}} \in \mathbb{R}^{N \times N}$, N is the number of nodes in global graph. Each entry $\mathbf{A}^{\mathcal{G}}_{i,j}$ of $\mathbf{A}^{\mathcal{G}}$ equals to 1 when u_j and u_j share the same post or $i = j$, otherwise $\mathbf{A}^{\mathcal{G}}_{i,j} = 0$. And $\mathbf{D}^{\mathcal{G}}_{i,i} = \sum_j^N \mathbf{A}^{\mathcal{G}}_{i,j}$. To learn the embedding of $\mathcal{G}$ via randomized tSVD-based SMF, we first define a proximity matrix $\mathbf{M}^{\mathcal{G}}$ as:

$$\mathbf{M}^{\mathcal{G}}_{i,j} = \begin{cases} ln p^{\mathcal{G}}_{i,j} - ln(\lambda \mathcal{N}^{\mathcal{G}}_{E,j}), & (u_i, u_j) \in E \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

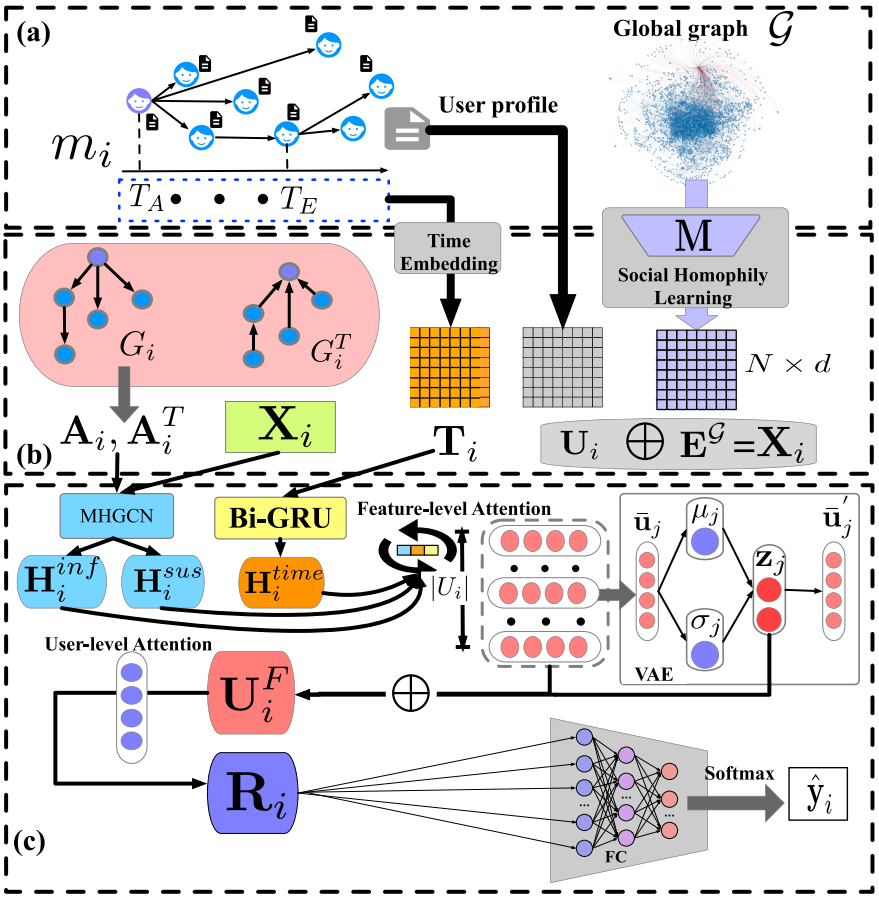


Fig. 3. Overview of **PLRD**: (a) input of **PLRD**; (b) preprocessing layer; (c) fine-grained user representation learning and rumor detection layer.

where $p_{i,j}^{\mathcal{G}} = \mathbf{A}_{i,j}^{\mathcal{G}} / \mathbf{D}_{i,i}^{\mathcal{G}}$ indicates the weight of (u_i, u_j) in E. $\mathcal{N}_{E,j}^{\mathcal{G}}$ are the negative samples connected with user u_j, which can be defined as $\mathcal{N}_{E,j}^{\mathcal{G}} \propto (\sum_{i:(i,j)\in E} p_{i,j})^{3/4}$ ($y \propto x$ means y and x are in a relation of directly proportional) (Levy & Goldberg, 2014). Then the goal of the global graph embedding is transformed to factorize the matrix $\mathbf{M}^{\mathcal{G}}$. Specifically, the step of approximate matrix factorization of $\mathbf{M}$[8] are follows the following three steps: (1) we first look for a matrix $\mathbf{Q} \in \mathbb{R}^{N\times d}$ with d orthonormal columns that let $\mathbf{M} \approx \mathbf{Q}\mathbf{Q}^T\mathbf{M}$; (2) suppose we found such matrix $\mathbf{Q}$, we define $\hat{\mathbf{M}} = \mathbf{Q}^T\mathbf{M} \in \mathbb{R}^{d\times N}$, which is quite smaller compare with the original matrix $\mathbf{M}$. Then we have $\hat{\mathbf{M}} = \mathbf{S}\mathbf{\Sigma}\mathbf{V}^T$, where $\mathbf{\Sigma}$ is the diagonal matrix with top-d singular values, and $\mathbf{S}, \mathbf{V} \in \mathbb{R}^{N\times d}$ are orthonormal matrices with d selected singular values; (3) finally, the factorize of $\mathbf{M}$ is approximate to $\mathbf{M} \approx \mathbf{Q}\mathbf{Q}^T\mathbf{M} = (\mathbf{Q}\mathbf{S})\mathbf{\Sigma}\mathbf{V}^T$ and the calculation of the output embeddings for the global graph is $\mathbf{E}^{\mathcal{G}} = \mathbf{Q}\mathbf{S}\mathbf{\Sigma}^{1/2}$ and $\mathbf{E}^{\mathcal{G}} = \{\mathbf{e}_j^{\mathcal{G}} | j \in [1, N]\} \in \mathbb{R}^{N\times d}$. That is, for each user u_j in the global graph, was allocated a relative latent embedding, i.e., $\mathbf{e}_j^{\mathcal{G}}$, and users with similar preferences and behavior (i.e., interested in the same posts) will have similar embeddings.

4.4. Users' influence and susceptibility learning

The role of each participant in the diffusion process of m_i has two types, i.e., sender and receiver (Wang et al., 2019). In previous diffusion research (Aral & Walker, 2012; Hoang & Lim, 2012), the role of the sender reflects a user's influence, that is, the ability to spread information to other users. On the opposite, the receiver role reflects the susceptibility of a user, i.e., the ability of a user to be infected by possible senders. In our work, we learn the influence and susceptibility for each user from the diffusion graph of m_i. However, in the original diffusion graph G_i, the information passed from a sender to receiver, so that modeling G_i can acquire influence for each user, and not efficient for susceptibility learning. To overcome this problem, we introduce an inverse diffusion graph G_i^T, which changes the direction of information propagation, i.e., from receiver to sender.

Inspired by the recent success of deep learning technologies in graph representation learning, such as graph convolutional network (GCN) (Defferrard et al., 2016; Kipf & Welling, 2017), graph attention network (GAT) (Veličković et al., 2018), etc. To

[8] For brevity, we ignore the superscript $\mathcal{G}$.

model the higher-order relationships among participants, we propose a multi-hop graph convolution layer (MHGCN) to extract user influence and susceptibility from G_i and G_i^T, respectively. The convolution kernel of MHGCN is defined as:

$$\mathbf{H} = g_\theta * \mathbf{X} = \sigma(\Big\|_{k\in\mathcal{O}} (\hat{\mathbf{A}}^{(k)}\mathbf{X}\mathbf{W}^{(k)})) \tag{2}$$

where $\Big\|_{k\in\mathcal{O}}$ represents the order-level concatenate. σ is a non-linear activation function such as ReLU. $\hat{\mathbf{A}}^{(k)}$ denotes the normalized adjacency matrix $\hat{\mathbf{A}} \in \mathbb{R}^{|U|\times|U|}$ multiplied by itself k times, $|U|$ is the number of nodes in graph, and $\mathcal{O}$ is a set of integer adjacency powers from 0 to K, K is the max-order. The calculation of normalized adjacency matrix is denoted as $\hat{\mathbf{A}} = \bar{\mathbf{D}}^{-1}\bar{\mathbf{A}}$, where $\bar{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, $\mathbf{I}$ is diagonal identity matrix. $\mathbf{X} \in \mathbb{R}^{N\times d_X}$ is the input graph signal, d_X is the dimension number. $\mathbf{W}^{(k)} \in \mathbb{R}^{d_X\times F}$ is the weight matrix for different order. Given the diffusion graph G_i and its inverse diffusion graph G_i^T, we have:

$$\begin{aligned}
\mathbf{X}_i &= \text{Concat}(\mathbf{E}_i, \mathbf{U}_i),\\
\mathbf{H}_i^{inf} &= \sigma(\Big\|_{k\in\mathcal{O}} (\hat{\mathbf{A}}_i^{(k)}\mathbf{X}_i\mathbf{W}_1^{(k)})),\\
\mathbf{H}_i^{sus} &= \sigma(\Big\|_{k\in\mathcal{O}} ((\hat{\mathbf{A}}_i^T)^{(k)}\mathbf{X}_i\mathbf{W}_2^{(k)})).
\end{aligned} \tag{3}$$

$\mathbf{E}_i = \{\mathbf{e}_j^{\mathcal{G}} | u_j \in U_i\}$, where $\mathbf{e}_j^{\mathcal{G}}$ is looked up through global graph embedding matrix $\mathbf{E}^{\mathcal{G}}$ by given user id, and $\mathbf{U}_i$ is the user characteristics matrix. $\mathbf{H}_i^{inf}, \mathbf{H}_i^{sus} \in \mathbb{R}^{|U_i|\times F}$ are user influence and user susceptibility, respectively.

4.5. Users' temporal learning

As mentioned in Definition 1 and Definition 3, the diffusion process for each m_i can be represented as a set of user-user-time tuples i,e., (u_i, u_j, t_j). Each retweet user u_j is associated with a specific timestamp. Modeling the timestamp information can extract the dynamic and temporal information for each participant user, which has been demonstrated to help rumor detection (Shu et al., 2020).

Assume that, the time window is $[0, T]$, we first split the time window into l disjoint time intervals, and compute the corresponding time interval for each retweet user u_j as $\text{pos} = \left\lfloor \frac{t_j - t_0}{T/l} \right\rfloor$, where t_0 is the timestamp for the source post user. Then, we use positional encoding introduced in the Transformer (Vaswani et al., 2017) to allocate initial embedding for each time interval.

$$\begin{aligned}
TP(\text{pos})_{2d} &= \sin \frac{\text{pos}}{10000^{2d/d_{time}}},\\
TP(\text{pos})_{2d+1} &= \cos \frac{\text{pos}}{10000^{2d/d_{time}}}.
\end{aligned} \tag{4}$$

where $\text{pos} \in [0, l)$ denotes the time interval each user fall into and d refers to dimension and d_{time} is the total dimensions of the time interval embedding. So that, as for a given tweet m_i, we acquire an initial time-embedding matrix denoted as $\mathbf{T}_i \in \mathbb{R}^{|U_i|\times d_{time}}$.

After that, we feed $\mathbf{T}_i$ into a Bi-directional GRU (Bi-GRU) (Chung et al., 2014) to learn the temporal information $\mathbf{H}_i^{time}$ for users.

$$\mathbf{H}_i^{time} = \text{Bi-GRU}(\mathbf{T}_i), \mathbf{H}_i^{time} \in \mathbb{R}^{|U_i|\times F} \tag{5}$$

4.6. Feature-level aggregation attention

After obtaining the multi-scale latent representation of users, i.e., $\mathbf{H}_i^{inf}$, $\mathbf{H}_i^{sus}$ and $\mathbf{H}_i^{time}$, we propose an attention-based method to capture the different importance among three types of representations. Let $\hat{\mathbf{u}}_j = \left[\mathbf{h}_j^{inf}, \mathbf{h}_j^{sus}, \mathbf{h}_j^{time}\right] \in \mathbb{R}^{3\times F}$ denote the learned feature set for user u_j. The attention $\mathbf{a}_j$ for $\hat{\mathbf{u}}_j$ is calculated as:

$$\begin{aligned}
\hat{\mathbf{u}}_j' &= \tanh\left(\hat{\mathbf{u}}_j \cdot \mathbf{w}_j\right),\\
\mathbf{a}_j &= \text{softmax}\left(\hat{\mathbf{u}}_j' \cdot \mathbf{w}_j'\right),
\end{aligned} \tag{6}$$

where $\mathbf{w}_j \in \mathbb{R}^{F\times F}$ and $\mathbf{w}_j' \in \mathbb{R}^{F\times 1}$ are weight matrices, $\mathbf{atten}_j \in \mathbb{R}^{3\times 1}$ is the learned attention. Then the aggregated feature vector for user u_j is $\bar{\mathbf{u}}_j = \hat{\mathbf{u}}_j \cdot \mathbf{a}_j$, where $\bar{\mathbf{u}}_j \in \mathbb{R}^F$. Finally, we get the fused user feature vector matrix as $\bar{\mathbf{U}}_i = \{\bar{\mathbf{u}}_j | j \in [1, |U_i|]\}$.

4.7. VAE-based uncertainty learning

In most of the existing works, the learned $\bar{\mathbf{U}}_i$ can be directly fed into a classification layer to predict the label of m_i. In our work, motivated by the ability of variational autoencoders (VAE) (Kingma & Welling, 2014) in coping with randomness and uncertainty, we employ VAE to capture the uncertainty in the learned user features. Let $f_{\text{Dec}}(\cdot)$, $f_{\text{Dec}}(\cdot)$ and $f_{\text{NN}}(\cdot)$ denote the encoder, decoder and neural network, respectively. Then the VAE-based uncertainty learning layer can be simply formalized as:

$$\begin{aligned}
\mathbf{z}_j &= f_{\text{ENC}}(\bar{\mathbf{u}}_j), \bar{\mathbf{u}}_j' = f_{\text{Dec}}(\mathbf{z}_j), j = 1, 2, \ldots, |U_i|.\\
\mu_j &= f_{\text{NN}}(\bar{\mathbf{u}}_j), \log\sigma_j^2 = f_{\text{NN}}(\bar{\mathbf{u}}_j), \mathbf{z}_j \sim \mathcal{N}(\mu_j, \sigma_j^2)
\end{aligned} \tag{7}$$

where $\bar{\mathbf{u}}_j'$ represents the reconstructed input features. $\mathbf{z}_j \in \mathbb{R}^{d_z}$ is the latent vector. Specifically, VAE gets μ and $\log\sigma^2$ from the encoder (we omit the subscript j for simplicity), and then samples latent representation $\mathbf{z}$ from Gaussian distribution via

reparameterization trick, where $\mathbf{z} = \mu + \sigma\epsilon$ and $\epsilon \sim \mathcal{N}(0, 1)$. Then the decoder takes the latent representation $\mathbf{z}$ as input, and try to reconstruct the original input feature. In general, the marginal log-likelihood of $\bar{\mathbf{u}} - \log p_\theta(\bar{\mathbf{u}}) = \log \int_{\mathbf{z}} p_\theta(\bar{\mathbf{u}}|\mathbf{z}) p(\mathbf{z}) d_{\mathbf{z}}$, which is intractable to compute effectively. Instead, adopts variational inference by defining a simple parametric distribution over the latent variables $q_\phi(\mathbf{z}|\bar{\mathbf{u}})$ (a.k.a. f_{Enc} parameterized by ϕ), and maximizing the evidence lower bound (ELBO) on the marginal log-likelihood of each observation:

$$\begin{aligned} \log p_\theta(\bar{\mathbf{u}}) &= \log \int_{\mathbf{z}} p_\theta(\bar{\mathbf{u}}|\mathbf{z}) p(\mathbf{z}) d_{\mathbf{z}} \\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\bar{\mathbf{u}})} \log[\frac{p_\theta(\bar{\mathbf{u}}, \mathbf{z})}{q_\phi(\mathbf{z}|\bar{\mathbf{u}})}] + \mathbb{KL}[q_\phi(\mathbf{z}|\bar{\mathbf{u}})||p_\theta(\mathbf{z}|\bar{\mathbf{u}})] \\ &\geq \mathbb{E}_{q_\phi(\mathbf{z}|\bar{\mathbf{u}})}[\log p_\theta(\bar{\mathbf{u}}, \mathbf{z}) - \log q_\phi(\mathbf{z}|\bar{\mathbf{u}})] \triangleq \text{ELBO}(\bar{\mathbf{u}}) \end{aligned} \tag{8}$$

To optimize the ELBO, we use a parametric inference network and reparameterization of $q_\phi(\mathbf{z}|\bar{\mathbf{u}})$ to alternatively maximize the following reformulation:

$$\begin{aligned} \text{ELBO}(\bar{\mathbf{u}}) &= \mathbb{E}_{q_\phi(\mathbf{z}|\bar{\mathbf{u}})}[\log p_\theta(\bar{\mathbf{u}}, \mathbf{z}) - \log q_\phi(\mathbf{z}|\bar{\mathbf{u}})] \\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\bar{\mathbf{u}})}[\log p_\theta(\mathbf{z}) + \log p_\theta(\bar{\mathbf{u}}|\mathbf{z}) - \log q_\phi(\mathbf{z}|\bar{\mathbf{u}})] \\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\bar{\mathbf{u}})} p_\theta(\bar{\mathbf{u}}|\mathbf{z}) - \mathbb{KL}[q_\phi(\mathbf{z}|\bar{\mathbf{u}})||p_\theta(\mathbf{z})] \end{aligned} \tag{9}$$

where $p_\theta(\bar{\mathbf{u}}|\mathbf{z})$ denotes the decoder and the first term of Eq. (9) is the reconstruction loss, which used to measure the likelihood value of the reconstructed features. The second term was the Kullback–Leibler (KL) divergence between the variational distribution $q_\phi(\mathbf{z}|\bar{\mathbf{u}})$ and the prior $p_\theta(\mathbf{z})$ (which is always ≥ 0). Therefore, the objective of maximizing ELBO of $\log p_\theta(\bar{\mathbf{u}})$ turns to minimize the Kullback–Leibler (KL) divergence. Through this VAE-based uncertainty learning layer, the learned latent representation for all users form a user latent representation matrix $\mathbf{U}_i^z = \{\mathbf{u}_j^z | j \in \mathbb{R}^{|U_i| \times d_z}\}$.

4.8. User-level aggregation attention

We concatenate $\bar{\mathbf{U}}_i$ and $\mathbf{U}_i^z$ at user-level to form a new user representation matrix $\mathbf{U}_i^F \in \mathbb{R}^{|U_i| \times (F + d_z)}$. Then we try to merge the user-level information to form an unique representation $\mathbf{R}_i$ for tweet m_i through attention sum-pooling operation:

$$\begin{aligned} \hat{a}_j &= \frac{\exp(\langle \mathbf{w}, \tanh(\mathbf{W}_a \mathbf{u}_j^F + \mathbf{b}_a) \rangle)}{\sum_{*:=1}^{|U_i|} \exp(\langle \mathbf{w}, \tanh(\mathbf{W}_a \mathbf{u}_*^F + \mathbf{b}_a) \rangle)}, \\ \mathbf{R}_i &= \sum_{j=1}^{N} \hat{a}_j \mathbf{u}_j^F \end{aligned} \tag{10}$$

where $\mathbf{W}_a \in \mathbb{R}^{(F+d_z) \times d}$, $\mathbf{b}_a \in \mathbb{R}^d$ and $\mathbf{w} \in \mathbb{R}^d$.

4.9. Rumor detection

Our ultimate goal is to predict the rumor label $\hat{y}_i$ of tweet m_i. We calculate this through feeding $\mathbf{R}_i$ into several fully connected layers and a softmax output layer, which is denoted as:

$$\hat{\mathbf{y}}_i = \text{softmax}\left(\text{FC}\left(\mathbf{R}_i\right)\right) \tag{11}$$

where $\hat{\mathbf{y}}_i$ is a vector of predicted probabilities of all rumor categories for the tweet m_i.

In implementation, we train PLRD to estimate all the model parameters by minimizing the *cross-entropy* of the predictions $\hat{\mathbf{Y}}$ and the ground truth labels $\mathbf{Y}$. The prediction loss is:

$$\mathcal{L}_{pre} = -\frac{1}{B} \sum_{i=1}^{B} \sum_{c=0}^{1} y_{i,c} \log \hat{y}_{i,c} \tag{12}$$

where $|B|$ is the batch size, $y_{i,c}$ and $\hat{y}_{i,c}$ are the ground truth and predicted results for the ith sample. That is, if the sample belongs to cth class, $\hat{y}_{i,c}$ is 1; otherwise it is 0.

The total loss of PLRD should take the ELBO into consideration, that is:

$$\mathcal{L} = \mathcal{L}_{pre} - \frac{1}{B} \sum_{i=1}^{B} \text{ELBO}(\bar{\mathbf{U}}_i) \tag{13}$$

During training, the well-known stochastic gradient descent is applied to update parameters. Specifically, we use the adaptive learning rate optimization algorithm Adam (Kingma & Ba, 2014) for model training. All hyper-parameters are tuned using the standard grid search for optimal results. The next section provides the details of the computational experiments.

Table 5
Overall performance comparison of rumor detection on Twitter15 and Twitter16. The best method is shown in **bold**, and the second best one is underlined. The number of retweets is 40.

Method	Twitter15				Twitter16			
	Acc	Pre	Rec	F1	Acc	Pre	Rec	F1
DTC	0.495	0.494	0.481	0.495	0.561	0.575	0.537	0.562
SVM-TS	0.519	0.519	0.518	0.519	0.693	0.692	0.691	0.692
GRU	0.580	0.544	0.545	0.544	0.554	0.514	0.516	0.515
TD-RvNN	0.678	0.671	0.674	0.672	0.661	0.632	0.641	0.636
PPC	0.691	0.674	0.686	0.679	0.655	0.632	0.651	0.641
dEFEND	0.738	0.658	0.661	0.654	0.702	0.637	0.638	0.631
Bi-GCN	0.748	0.731	0.759	0.745	0.711	0.709	0.710	0.716
GCAN	0.875	0.825	0.829	0.825	0.823	0.803	0.841	0.822
GCAN-Text	0.683	0.705	0.652	0.678	0.664	0.716	0.579	0.648
PLRD	**0.934**	**0.928**	**0.929**	**0.927**	**0.875**	**0.876**	**0.874**	**0.855**
Improvement	**8.98%**	**12.5%**	**12.1%**	**12.4%**	**6.32%**	**9.09%**	**3.92%**	**4.01%**

4.10. Computational complexity analysis

In this section, we give a brief analysis of the computational complexity of PLRD. (1) The complexity for social homophily learning from global graph: as analyzed in Zhang et al. (2019), the overall complexity of this layer is $\mathcal{O}(d^2|U| + |E|)$, where d, $|U|$ and $|E|$ are the dimensions of user social homophily, number of nodes and edges in global graph, respectively. (2) The complexity for users' influence and susceptibility learning: we use a multi-hop graph convolutional layer to learn the users' influence and susceptibility (cf. Eq (2)). Recall that, the dimensions of input and output are d_X and F, respectively, the max-order is K, and the normalized adjacency matrix $\hat{\mathbf{A}}$ is a sparse matrix with m nonzero elements. Therefore, for a single MHGCN layer, the computational complexity is $\mathcal{O}(F \times K \times m \times d_X)$. (3) The other parts of the PLRD are implemented by GRUs and MLPs. The time and space complexity are related to the input dimensions of latent variables. Since the users' social homophily are computed in preprocessing phase, the computational complexity of whole PLRD is therefore $\mathcal{O}(F \times K \times m \times d_X)$.

5. Evaluation

Following the design science paradigm, we rigorously evaluate our proposed PLRD framework and demonstrate its practical utility through quantitative experiments.

5.1. Evaluation metrics and baselines

In our work, we use Accuracy (ACC), Precision (Pre), Recall (Rec), and F1 as the evaluation protocols to measure the models' performance. In particular, ACC measures the proportion of correctly classified tweets, while F1 is the harmonic mean of the precision and recall.

We rigorously compare our method with a battery of baselines, they are:

- **DTC** (Castillo et al., 2011): A decision tree-based classification model that combines manually engineered characteristics of tweets to compute the information credibility.
- **SVM-TS** (Ma et al., 2015): A linear SVM-based time series model, which can capture the variation of a broad spectrum of social context information over time by converting the continuous-time stream into fixed time intervals.
- **GRU** (Ma et al., 2016): An RNN-based model has been employed to learn the sequential cascading effect of tweets with high-level feature representations extracted from relevant posts over time.
- **TD-RvNN** (Ma et al., 2018b): A tree-structured RNN model for rumor detection, which embeds hidden indicative signals in the tree-structures and explores the importance of comments for rumor detection.
- **PPC_RNN+CNN** (Liu & Wu, 2018): An early-stage rumor detection model through classifying news propagation paths with RNN and CNN, which learns the rumor representations through the characteristics of users and source tweets (for brevity, the model name is abbreviated to PPC).
- **dEFEND** (Shu et al., 2019a): A co-attention-based fake news detection model that exploits both news contents and user comments for fake news detection.
- **Bi-GCN** (Bian et al., 2020): A GCN-based model exploiting the bi-directional propagation structures and comments for rumor detection.
- **GCAN** (Lu & Li, 2020): A co-attention network that detects true and false rumors based on the content of the source tweet and its propagation-based users.

Table 6
Overall performance comparison of rumor detection on Science. The best method is shown in **bold**. The number of retweets is 40.

Method	Science			
	Acc	Pre	Rec	F1
PPC	0.655	0.649	0.568	0.606
GCAN-Text	0.671	0.646	0.622	0.634
PLRD	**0.768**	**0.727**	**0.814**	**0.768**

5.2. Experimental setup

We implement DTC with Weka,[9] SVM-based models with scikit-learn,[10] and other neural network-based models with Tensorflow.[11] All baselines follow the parameter settings in the original papers.

For PLRD, the learning rate is initialized at 0.001 and gradually decreases as the training proceeds. The embedding size for social homophily set d to 40. As for time-embedding, we set the total number of time intervals to be 100 and each interval represents 10 minutes. Retweets with a latency of more than 1,000 min would fall into the last time interval. The size of time-embedding d_{time} set to 50. The hidden size F of user influence, susceptibility, and temporal information are setting to 64; the hidden size d_z of the VAE-based uncertainty learning layer is set to 32. The batch size is set to 32 for Twitter15/16, 128 for Science and 16 for RumourEval19, and the training process is iterated upon for 500 epochs but would be stopped earlier if the validation loss does not decrease after 10 epochs. And we randomly choose 70% data for training and the rest of 10% and 20% for validation and testing. The experiments are conducted on a machine with an Intel i7-6700 3.40 GHZ CPU and a single NVIDIA GeForce GTX 2080Ti. The time cost for model training is less than 10 minutes on all datasets used in this work.

5.3. Study on Twitter15/16

Table 5 summarizes the overall performance on Twitter15 and Twitter16 datasets. The last two rows show the performance of the complete version of our model PLRD and the improvement percentage compare with the second-best method, which basically yields much better performance than the other baseline methods across all metrics. Note that we conduct a McNemar's test (Dietterich, 1998) between our PLRD and the best baseline based on the prediction results on the testing set. The p-values are $p < 0.001$ on both Twitter15 and Twitter16. Therefore, we can conclude that the performance between PLRD and GCAN exists statistical significance.

We make the following additional observations. *O1:* The feature-based approaches — DTC and SVM-TS use hand-crafted features based on the overall statistics of tweets, which perform poorly. These two methods not sufficient to capture the generalizable features associated with tweets and the diffusion process. Notably, SVM-TS achieves comparatively better performance than DTC because it utilizes an extensive set of features and focuses on retweets' temporal features. *O2:* Deep learning-based models achieve better performance than feature-based methods. GRU is the first deep-learning-based method for rumor detection, which performs worst among deep-learning-based baselines because it only relies on the temporal-linguistics features of the post sequence but ignores other useful information such as diffusion structures and user profiles. Both TD-RvNN and PPC outperform GRU, which indicates the effectiveness of modeling the propagation structure and temporal information in rumor detection. Moreover, PPC proves user profile features as important as text features for rumor detection. dEFEND utilize a co-attention mechanism to learn the correlation between news contents and user comments, which performs better than TD-RvNN and PPC but worse than Bi-GCN and GCAN. Bi-GCN and GCAN claim that they can learn structure information from graphs, and their performance indeed exceeds other baseline methods. However, Bi-GCN constructs the structural tree based on the replies, which cannot reflect the full process of rumor diffusion. As for GCAN, it captures the similarity between users rather than propagation structural features. According to the results, GCAN performs much better than Bi-GCAN, because it takes both text information and user profiles into consideration. By comparing GCAN with its variants GCAN-Text, we can find that after removing text information, the performance of GCAN remarkably decrees, demonstrating that GCAN is not efficient to capture user-related features. *O3:* PLRD consistently outperforms all baselines on both Twitter15 and Twitter16. Compare to the best baseline method Bi-GCN, PLRD learns rumor representation from a participant-level without any text information, demonstrate the primary motivations of this work – i.e., users are the main contributor to the rumor propagation.

5.4. Study on science

In this section, we conduct an experiment on Science dataset. Specifically, from the original Science dataset, we first filter out (1) the tweets with less than 10 retweets and more than 100; and (2) the tweets' with a diffusion period exceed 24 hours. After that, we have 3,493 tweets in total, and the processed dataset is still highly imbalanced, e.g., only 610 tweets were labeled as "non-rumor", while the majority (i.e., 2,883) were classified as "rumor". We randomly select 1,000 items from the rumor set to make sure the

[9] https://www.cs.waikato.ac.nz/ml/weka/.
[10] https://scikit-learn.org/.
[11] https://www.tensorflow.org/.

Table 7
Overall performance comparison of rumor detection on RumourEval19. The best method is shown in **bold**. The result of the "Base" model has referenced the best method from the paper (Gorrell et al., 2019). "w/o Text" means without textual features. The number of comments is 100.

Method	RumourEval19			
	Acc	Pre	Rec	Macro-F1
Base	–	0.596	0.603	0.577
TD-RvNN	0.667	0.641	0.673	0.615
Bi-GCN	0.734	0.733	0.735	0.661
PLRD	**0.813**	**0.826**	**0.885**	**0.788**
PLRD w/o Text	0.75	0.806	0.842	0.692

number of tweets labeled as non-rumors is 50% of the rumors, and finally, we get a new experiment dataset with 1000 rumors and 610 non-rumors. Table 6 summarizes the performance comparison between PLRD and two state-of-the-art propagation-based baselines, i.e., PPC and GCAN-Text.[12] We can observe that our PLRD outperforms both PPC and GCAN-Text on all metrics, which indicates that our model is more effective and stable in extracting diffusion patterns of users even without any textual information.

5.5. Study on RumourEval19

Since the RumorEval19 dataset has rich textual features, in this section we conduct an experiment on RumourEval19 to test the PLRD performance when including textual features. Specifically, we first use a pre-trained model – BERTweet (Nguyen et al., 2020) – to generate the tweet embedding for each source tweet and its corresponding comments, and then concatenate the source tweet embedding and comments embedding to form a textual embedding matrix $\mathbf{C} \in \mathbb{R}^{|U_i| \times d_{text}}$ for each tweet. Finally, we combine $\mathbf{C}$ with $\mathbf{X}$ to form the input of PLRD. For a comparison, we choose TD-RvNN and Bi-GCN as baselines, and also provide the result of the best method in (Gorrell et al., 2019) denotes as "Base". From Table 7, we can find that under the situation of unbalanced label distribution, our PLRD can still achieve competitive performance compared with other baselines on rumor detection. After deleting the textual features, the performance of PLRD slightly drops, which demonstrate that textual features are powerful and can help improve the model performance.

5.6. Ablation study

In this section, we conduct an ablation study on Twitter15 and Twitter16 to explore the effect of each component in PLRD. Towards that, we derive the following variants of PLRD:

- **w/o user profiles (UP)**: In "**w/o UP**", we do not consider the user profile characteristics, which means we do not use $\mathbf{U}_i$ and only keep $\mathbf{E}_i$ as input features.
- **w/o social homophily (SH)**: In "**w/o SH**", we ignore the social homophily of users, which means we do not use $\mathbf{E}_i$ and only keep $\mathbf{U}_i$ as input features.
- **w/o user influence (UI)**: In "**w/o UI**", we do not capture user influence, which means ignore $\mathbf{H}_i^{inf}$.
- **w/o user susceptibility (US)**: In "**w/o US**", we do not capture user susceptibility, which means ignore $\mathbf{H}_i^{sus}$.
- **w/o user temporal (UT)**: In "**w/o UT**", we do not take the user temporal information into consideration, which means ignore $\mathbf{H}_i^{time}$.
- **w/o graph information (GI)**: In "**w/o GI**", we do not utilize any information from the global graph and the diffusion graph, which means we ignore users' social homophily, influence, susceptibility and only keep the users' temporal learning component. The input of this part is the concatenation of user features and temporal information.
- **w/o feature uncertainty (FU)**: In "**w/o FU**", we remove the VAE-based uncertainty learning layer and use $\bar{\mathbf{U}}_i$ directly.
- **w/o feature-level attention (FA)**: In "**w/o FA**", we remove the feature-level aggregation attention in PLRD and concatenate the different user features directly, i.e., $\bar{\mathbf{U}}_i = \text{concat}(\mathbf{H}_i^{inf}, \mathbf{H}_i^{sus}, \mathbf{H}_i^{time}) \in \mathbb{R}^{|U_i| \times 3F}$.
- **w/o user-level attention (UA)**: In "**w/o UA**", we do not allocate different importance for each user and directly use a sum-pooling to form the rumor representation.

The results, shown in Table 8, indicate that the original PLRD outperforms these variants in terms of all metrics. From Table 8, we can observe that: (1) Both user profile features (w/o UP) and social homophily ('w/o SH) are reliable inputs of our model that because user profiles can be used to identify an individual, and social homophily can reflect the user preference. (2) User influence (w/o UI), susceptibility (w/o US), and temporal features (w/o UT) are indispensable for rumor detection. (3) The result of "w/o GI" performs worst among all variants, demonstrating that graph data (global graph and diffusion graph) provide considerable meaningful features, and are thus indispensable in rumor detection. (4) The fact that "w/o FU" provides lower performance compare

[12] The Science dataset only provides anonymous user profile characteristics and propagation threads. No textual features are available. So we compare with PPC and GCAN-Text.

Table 8
Performance comparison between PLRD and its variants.

Method	Twitter15				Twitter16			
	Acc	Pre	Rec	F1	Acc	Pre	Rec	F1
w/o UP	0.853	0.842	0.838	0.828	0.838	0.858	0.847	0.840
w/o SH	0.906	0.894	0.910	0.896	0.802	0.821	0.786	0.794
w/o UI	0.868	0.863	0.871	0.859	0.800	0.859	0.871	0.867
w/o US	0.841	0.877	0.857	0.864	0.781	0.794	0.775	0.782
w/o UT	0.873	0.914	0.935	0.920	0.795	0.792	0.802	0.788
w/o GI	0.806	0.755	0.811	0.782	0.758	0.716	0.732	0.724
w/o FU	0.913	0.911	0.914	0.911	0.854	0.847	0.859	0.848
w/o FA	0.811	0.842	0.843	0.848	0.790	0.831	0.837	0.816
w/o UA	0.896	0.906	0.877	0.886	0.854	0.838	0.831	0.829
PLRD	**0.934**	**0.928**	**0.929**	**0.927**	**0.875**	**0.876**	**0.874**	**0.855**

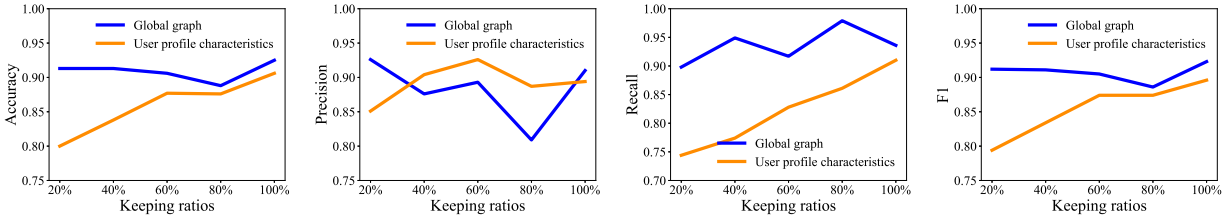


Fig. 4. Evaluations on random removing different proportions of edges in the global graph and random masking different proportions of user characteristics.

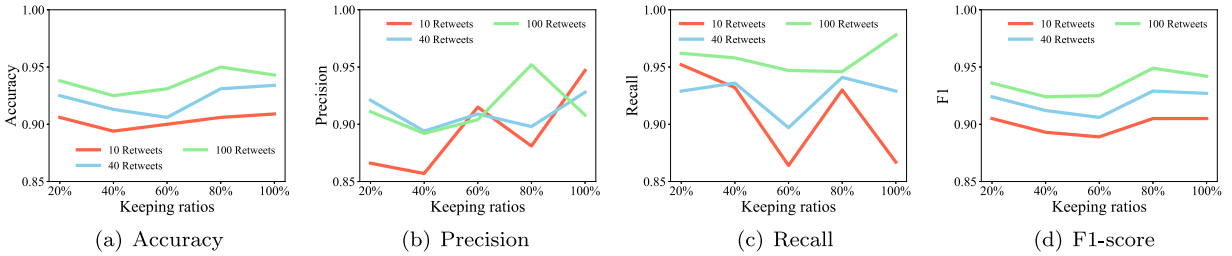


Fig. 5. Evaluations on random removal of diffusion links based on Twitter15.

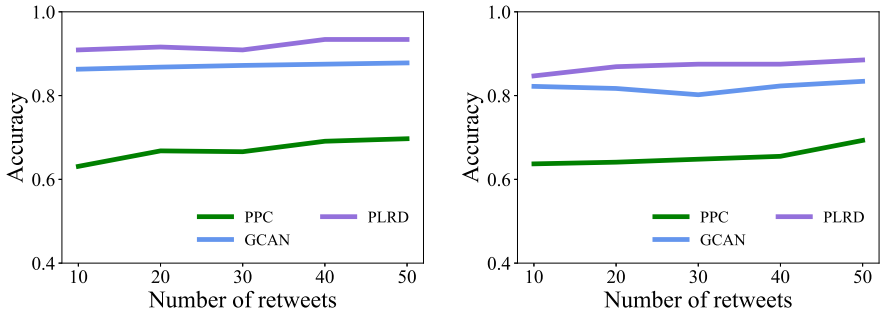
with PLRD, reflects the benefit of modeling the feature uncertainty. (5) The two attention-based aggregation layers, i.e., feature-level aggregation attention (w/o FA) and user-level aggregation attention (w/o UA), play crucial roles in detecting rumors. Especially the feature-level aggregation attention (w/o FA), after removing it, the performance remarkably decreases, suggesting that distinguishing the importance of different scale of user features can improve detection performance. Similarly, "w/o UA" demonstrates that different users play different roles in rumor spreading.

5.7. *Privacy-preserving study*

In this section, we conduct experiments on Twitter15 dataset to test our PLRD's performance on privacy-preserving scenarios which can be summarized as: (1) random removing different proportions of edges in the global graph $\mathcal{G}$; (2) random masking different proportions of user characteristics $\mathbf{u}$, and (3) random removing different proportions of edges in the diffusion graph G. Note that, in scenario (1) and (2) we only keep related features as inputs, i.e., in scenario (1), we only use the user homophily $\mathbf{E}$ generated from the global graph as input, and in scenario (2) we only keep user characteristic $\mathbf{U}$ as input. And both of these two scenarios are testing on 40 retweets.

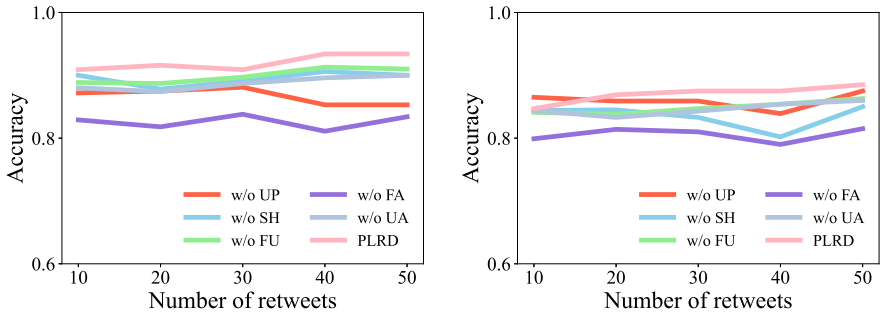
plots the performance of PLRD in scenarios (1) and (2), which shows that even though we only keep 20% of edges in the global graph, the PLRD still achieves 90% accuracy. However, when masking 80% of user characteristics, the performance of PLDR drops significantly.

Fig. 4(d) shows the performance of PLRD with the different numbers of retweets in scenario (3). We find that when we only observe few retweets, e.g., 10 and 40, the performance of the model decreases as the removal of edges in the diffusion graph. In contrast, when the number of observation retweets is sufficient enough such as 100, dropping a few edges would help improve the model performance. The reasons behind is that: (1) with the increase of observed retweets, there is a great possibility to introduce



(a) Early 50 retweets (Twitter15). (b) Early 50 retweets (Twitter16).

Fig. 6. Evaluations on early rumor detection.



(a) Early 50 retweets (Twitter15). (b) Early 50 retweets (Twitter16).

Fig. 7. Evaluations on early rumor detection among variants of PLRD.

noise into the graph, which can be eliminated with random removal of graph edges; and (2) randomly drop a few of graph edges is widely used as a data augmentation method in graph representation learning field (Rong et al., 2020; Zhao et al., 2020), which can improve model generalization and overcome the overfitting and over-smooth issues of graph neural networks.

5.8. Early detection

Another critical goal of rumor detection is to detect rumors as early as possible that is essential to stop their spread in a timely fashion. Now we investigate the performance of models on identifying rumors at an early stage. Here, we consider the early 50 retweets.

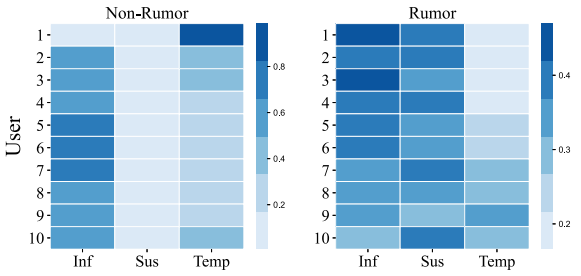
Fig. 5(d) shows the performance comparison on early-stage detection between our PLRD and the selected baselines. Note that we omit the feature-based approaches (i.e., DTC and SVM-TS) and GRU since they did not show comparable performance, especially on early rumor detection. Moreover, we also ignore TvRvNN, dEFEND, and Bi-GCN, because these methods are built on the replies that may not exist in the early-stage. We observe that PLRD performs better than PPC and GCAN, especially when there are only a few observations. PLRD needs a short time to identify the misinformation because PLRD learns the rumor representation from a participant-level and fuses users' multi-scale knowledge, such as user influence, user susceptibility, user temporal information, etc.

We also investigate the time-varying performance between PLRD and its variants. Specifically, we choose "w/o UP", "w/o SH", "w/o FU", "w/o FA" and "'w/o UA' as the comparison methods. The results show in Fig. 6(b). We find that the performance of PLRD surpasses all variants, and with the number of retweets increased, the accuracy of all methods grow to saturation.

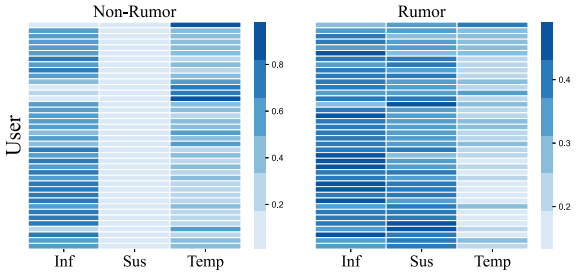
5.9. Interpretability analysis

The above ablation studies have shown the superiority of each component in PLRD. In this section, we provide more in-depth insights by feature visualizing.

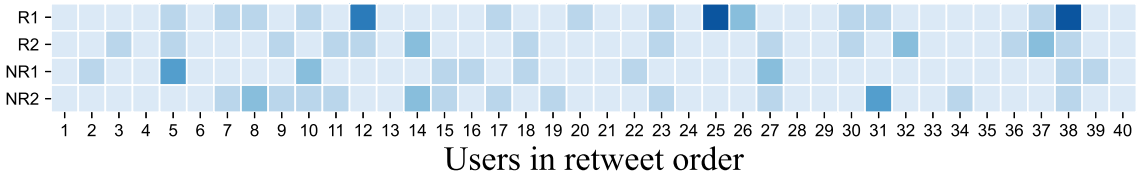
Fig. 7(b) plots the importance of the feature-level aggregation layer and user-level aggregation layer. As for the feature-level aggregation attention, we randomly selected two different types of tweets in Twitter15 and plotted the importance of the different features. Figs. 8(a) and 8(b) show the results of previous 10 and 40 retweet users, respectively. Overall, we find that (1) the three types of features for each user have different importance; (2) attention distribution varies between rumor and non-rumor. Specifically, as



(a) Feature-level aggregation attention (10 retweet user).



(b) Feature-level aggregation attention (40 retweet user).



(c) User-level aggregation attention.

Fig. 8. Attention visualization of PLRD. "Inf": influence; "Sus": susceptibility; "Temp": temporal.

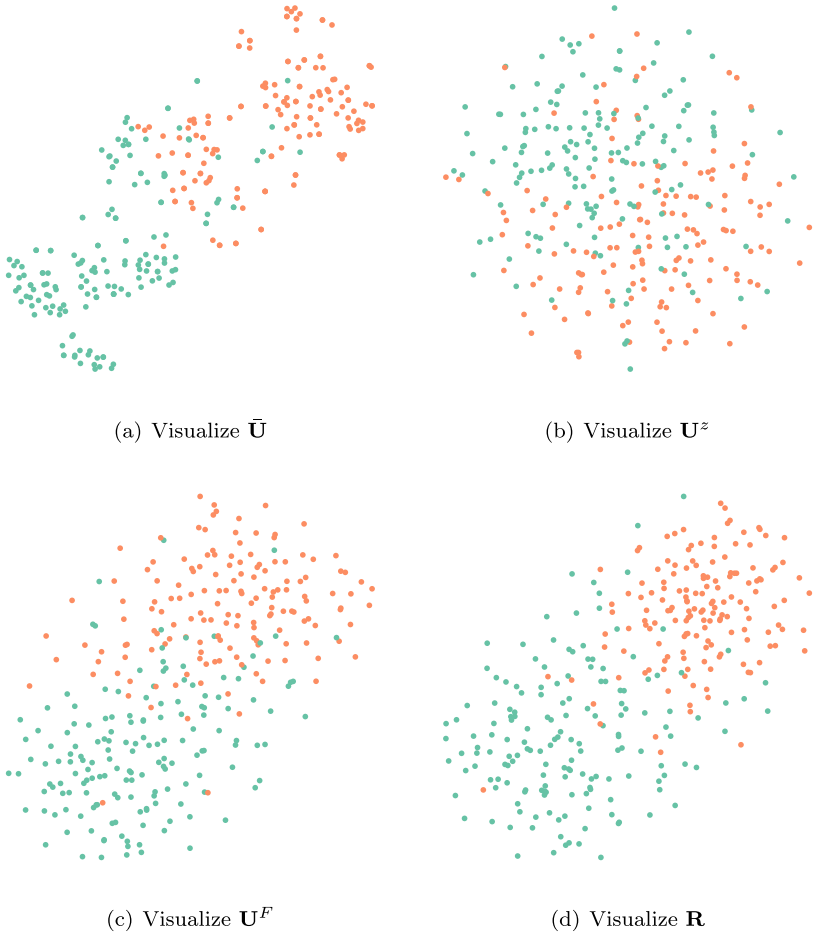
for rumor tweets, participants try to affect others, while themselves are easier to expose in the misleading tweets. In contrast, in non-rumor tweets, participants are more influential compare with the susceptibility. Moreover, temporal information plays a crucial role in detecting both rumor and non-rumor. In Fig. 8(c), we investigate the role of the retweet users at the very beginning of the diffusion. As shown, the later users are more critical in rumor spreading, which confirms the hypothesis that rumors can spread deeper than non-rumors (Vosoughi et al., 2018).

Moreover, to have an intuitive explanation regarding the superiority of each component in PLRD, we plot the learned latent representations (i.e., $\bar{\mathbf{U}}$, $\mathbf{U}^z$, $\mathbf{U}^F$ and $\mathbf{R}$) using t-SNE (Van Der Maaten, 2014). Each point in the plot represents a tweet in the test set (tweets with similar latent vectors are closer in the plot), and different colors refer to different labels, i.e., green represents non-rumor, orange represents rumor. From Fig. 9(a), we see clear clustering phenomena by $\bar{\mathbf{U}}$. These latent vectors already can be used to predict directly. In contrast, Fig. 9(b) "smoothes" this clustering effect by modeling the feature uncertainty, which should help explore more possibilities. From Figs. 9(c)–9(d), we find that the model learned more suitable latent representations for prediction after a user-level attention layer.

6. Conclusion and discussion

In this work, we first provide empirical evidence that all participants in the diffusion chains of rumors exhibit different patterns than participants in the diffusion chains of non-rumors. Based on these findings, we propose a novel fine-grained all-participant level rumor detection model, named PLRD (*P*articipant-*L*evel *R*umor *D*etection). Specifically, PLRD learns fine-grained user representations, i.e., user influence, user susceptibility, and user temporal from the propagation threads of a given post, and merges the learned features to form a unique rumor representation through a feature-level attention layer and a user-level attention layer. Moreover, a variational autoencoder used to capture uncertainty from features further improves the learned rumor representation. Compared with existing rumor detection methods, PLRD makes predictions only based on user-level features learned from the diffusion process of posts, which overcomes the problem of overemphasizing the text features. We conduct experiments on four real-world datasets, Twitter15, Twitter16, Science and RumourEval19. The experiment results not only demonstrate that our model significantly outperforms the baselines regarding effective early detection, but also supports the hypothesis that the combination of various user information at a participant level in a diffusion chain will improve the performance of rumor discovery. Besides, our ablation study further demonstrates that each part in our model is indispensable for rumor detection.

Our work also has several limitations that can be addressed in the future. For example, the experimental datasets we used are small, and a small portion of user-profiles are no longer available. This limitation can be solved by developing a stable fake news collector (Shu et al., 2020). Although our method shows a strong ability to make predictions without any text information, undeniable, when the tweet just posts and without any retweet, the text features are still powerful and indispensable, joint the text features into our model is one of our future tasks. Besides, users' stance (Dungs et al., 2018; Li et al., 2019), sentiment (Qin et al., 2021), and intentions are critical in identifying misinformation, which needs careful consideration of why a particular user is involved in retweeting a tweet. Finally, we will extend our model to other downstream tasks, such as rumor spreader prediction (Rath et al., 2021), user interest prediction (Zarrinkalam et al., 2017, 2018), community prediction (Fani et al., 2020), recommendation (Pourali et al., 2019), and information cascades modeling (Chen et al., 2019), etc.



(a) Visualize $\bar{\mathbf{U}}$ (b) Visualize $\mathbf{U}^z$

(c) Visualize $\mathbf{U}^F$ (d) Visualize $\mathbf{R}$

Fig. 9. Visualization of the learned latent representation on Twitter15 using t-SNE. Each point is a sample from the test set. The color green represents non-rumor, and the orange one represents rumor. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

CRediT authorship contribution statement

Xueqin Chen: Conceptualization, Methodology, Experiments, Writing. **Fan Zhou:** Funding acquisition, Conceptualization, Writing - review & editing. **Fengli Zhang:** Validation, Visualization, Writing - review & editing. **Marcello Bonsangue:** Visualization, Resources, Writing - review & editing.

Acknowledgments

This work was supported by National Natural Science Foundation of China (Grant No. 62072077 and No. 61602097), Sichuan Regional Innovation Cooperation Project (Grant No. 2020YFQ0018).

References

Allcott, H., & Gentzkow, M. (2017). Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, *31*(2), 211–236, URL: https://doi.org/10.1257/jep.31.2.211.

Aral, S., & Walker, D. (2012). Identifying influential and susceptible members of social networks. *Science*, 337–341, URL: https://doi.org/10.1126/science.1215842.

Bian, T., Xiao, X., Xu, T., Zhao, P., Huang, W., Rong, Y., & Huang, J. (2020). Rumor detection on social media with bi-directional graph convolutional networks. In *Proceedings of the thirty-fourth AAAI conference on artificial intelligence* (pp. 549–556). URL: https://doi.org/10.1609/aaai.v34i01.5393.

Cao, S., Lu, W., & Xu, Q. (2015). GraRep: Learning graph representations with global structural information. In *Proceedings of the 24th ACM international on conference on information and knowledge management* (pp. 891–900). URL: https://doi.org/10.1145/2806416.2806512.

Castillo, C., Mendoza, M., & Poblete, B. (2011). Information credibility on Twitter. In *Proceedings of the 20th international conference on world wide web* (pp. 675–684). URL: https://doi.org/10.1145/1963405.1963500.

Chen, X., Zhou, F., Zhang, K., Trajcevski, G., Zhong, T., & Zhang, F. (2019). Information diffusion prediction via recurrent cascades convolution. In *IEEE 35th international conference on data engineering* (pp. 770–781). URL: https://doi.org/10.1109/ICDE.2019.00074.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 workshop on deep learning*.

Defferrard, M., Bresson, X., & Vandergheynst, P. (2016). Convolutional neural networks on graphs with fast localized spectral filtering. In *Proceedings of the 30th international conference on neural information processing systems* (pp. 3844–3852).

Derczynski, L., Bontcheva, K., Liakata, M., Procter, R., Hoi, G. W. S., & Zubiaga, A. (2017). SemEval-2017 Task 8: RumourEval: Determining rumour veracity and support for rumours. In *Proceedings of the 11th international workshop on semantic evaluation* (pp. 69–76). URL: https://doi.org/10.18653/v1/S17-2006.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, *10*(7), 1895–1923, URL: https://doi.org/10.1162/089976698300017197.

DiFonzo, N., & Bordia, P. (1997). Rumor and prediction: Making sense (but losing dollars) in the stock market. *Organizational Behavior and Human Decision Processes*, *71*(3), 329–353, URL: https://doi.org/10.1006/obhd.1997.2724.

Dungs, S., Aker, A., Fuhr, N., & Bontcheva, K. (2018). Can rumour stance alone predict veracity? In *Proceedings of the 27th international conference on computational linguistics* (pp. 3360–3370). URL: https://www.aclweb.org/anthology/C18-1284.

Fani, H., Bagheri, E., & Du, W. (2020). Temporal latent space modeling for community prediction. In J. M. Jose, E. Yilmaz, J. Magalhães, P. Castells, N. Ferro, M. J. Silva, & F. Martins (Eds.), *Advances in information retrieval* (pp. 745–759).

Gorrell, G., Kochkina, E., Liakata, M., Aker, A., Zubiaga, A., Bontcheva, K., & Derczynski, L. (2019). SemEval-2019 task 7: RumourEval, determining rumour veracity and support for rumours. In *Proceedings of the 13th international workshop on semantic evaluation* (pp. 845–854). URL: https://doi.org/10.18653/v1/S19-2147.

Gupta, M., Zhao, P., & Han, J. (2012). Evaluating event credibility on twitter. In *Proceedings of the 2012 SIAM international conference on data mining* (pp. 153–164). URL: https://doi.org/10.1137/1.9781611972825.14.

Halko, N., Martinsson, P. G., & Tropp, J. A. (2011). Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, *53*, 217–288, URL: https://doi.org/10.1137/090771806.

Hoang, T.-A., & Lim, E.-P. (2012). Virality and susceptibility in information diffusions. In *Proceedings of the sixth international conference on weblogs and social media*.

Jin, Z., Cao, J., Zhang, Y., Zhou, J., & Tian, Q. (2017). Novel visual and statistical image features for microblogs news verification. *IEEE Transactions on Multimedia*, *19*(3), 598–608, URL: https://doi.org/10.1109/TMM.2016.2617078.

Kelman, H. C. (1958). Compliance, identification, and internalization three processes of attitude change. *Journal of Conflict Resolution*, *2*(1), 51–60, URL: https://doi.org/10.1177/002200275800200106.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Kingma, D. P., & Welling, M. (2014). Auto-encoding variational bayes. In *International conference on learning representations*.

Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In *International conference on learning representations*.

Kwon, S., Cha, M., Jung, K., Chen, W., & Wang, Y. (2013). Prominent features of rumor propagation in online social media. In *2013 IEEE 13th international conference on data mining* (pp. 1103–1108). URL: https://doi.org/10.1109/ICDM.2013.61.

Levy, O., & Goldberg, Y. (2014). Neural word embedding as implicit matrix factorization. In *Proceedings of the 27th international conference on neural information processing systems* (pp. 2177–2185).

Li, J., Ni, S., & Kao, H. Y. (2020). Birds of a feather rumor together? Exploring homogeneity and conversation structure in social media for rumor detection. *IEEE Access*, *8*, 212865–212875, URL: https://doi.org/10.1109/ACCESS.2020.3040263.

Li, Q., Zhang, Q., & Si, L. (2019). Rumor detection by exploiting user credibility information, attention and multi-task learning. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 1173–1179). URL: https://doi.org/10.18653/v1/P19-1113.

Liu, Y., & Wu, Y.-F. B. (2018). Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Proceedings of the thirty-second AAAI conference on artificial intelligence* (pp. 354–361).

Lu, Y.-J., & Li, C.-T. (2020). GCAN: Graph-aware co-attention networks for explainable fake news detection on social media. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 505–514). URL: https://doi.org/10.18653/v1/2020.acl-main.48.

Ma, J., Gao, W., Mitra, P., Kwon, S., Jansen, B. J., Wong, K.-F., & Cha, M. (2016). Detecting rumors from microblogs with recurrent neural networks. In *Proceedings of the twenty-fifth international joint conference on artificial intelligence* (pp. 3818–3824).

Ma, J., Gao, W., Wei, Z., Lu, Y., & Wong, K. (2015). Detect rumors using time series of social context information on microblogging websites. In *Proceedings of the 24th ACM international on conference on information and knowledge management* (pp. 1751–1754). URL: https://doi.org/10.1145/2806416.2806607.

Ma, J., Gao, W., & Wong, K.-F. (2017). Detect rumors in microblog posts using propagation structure via kernel learning. In *Proceedings of the 55th annual meeting of the association for computational linguistics* (pp. 708–717). URL: https://doi.org/10.18653/v1/P17-1066.

Ma, J., Gao, W., & Wong, K. (2018). Detect rumor and stance jointly by neural multi-task learning. In *Companion proceedings of the the web conference 2018* (pp. 585–593). URL: https://doi.org/10.1145/3184558.3188729.

Ma, J., Gao, W., & Wong, K.-F. (2018). Rumor detection on Twitter with tree-structured recursive neural networks. In *Proceedings of the 56th annual meeting of the association for computational linguistics* (pp. 1980–1989). URL: https://doi.org/10.18653/v1/P18-1184.

McPherson, M., Smith-Lovin, L., & Cook, J. M. (2001). Birds of a feather: Homophily in social networks. *Review of Sociology*, *27*, 415–444, URL: https://doi.org/10.1146/ANNUREV.SOC.27.1.415.

Nguyen, D. Q., Vu, T., & Tuan Nguyen, A. (2020). BERTweet: A pre-trained language model for English Tweets. In *Proceedings of the 2020 conference on empirical methods in natural language processing: System demonstrations* (pp. 9–14). URL: https://doi.org/10.18653/v1/2020.emnlp-demos.2.

Nickerson, R. S. (1998). Confirmation bias: A ubiquitous phenomenon in many guises. *Review of General Psychology*, *2*(2), 175–220, URL: https://doi.org/10.1037/1089-2680.2.2.175.

Pourali, A., Zarrinkalam, F., & Bagheri, E. (2019). Neural embedding features for point-of-interest recommendation. In *2019 IEEE/ACM international conference on advances in social networks analysis and mining* (pp. 657–662). URL: https://doi.org/10.1145/3341161.3343672.

Qin, L., Li, Z., Che, W., Ni, M., & Liu, T. (2021). Co-GAT: A co-interactive graph attention network for joint dialog act recognition and sentiment classification. In *The thirty-fifth AAAI conference on artificial intelligence*.

Rath, B., Morales, X., & Srivastava, J. (2021). SCARLET: Explainable attention based graph neural network for fake news spreader prediction. In *The 25th pacific-asia conference on knowledge discovery and data mining*.

Rong, Y., Huang, W., Xu, T., & Huang, J. (2020). DropEdge: Towards deep graph convolutional networks on node classification. In *International conference on learning representations*.

Sankar, A., Zhang, X., Krishnan, A., & Han, J. (2020). Inf-VAE: A variational autoencoder framework to integrate homophily and influence in diffusion prediction. In *Proceedings of the 13th international conference on web search and data mining* (pp. 510–518). URL: https://doi.org/10.1145/3336191.3371811.

Shore, J., Baek, J., & Dellarocas, C. (2018). Network structure and patterns of information diversity on Twitter. *MIS Quarterly*, *42*(3), 849–872, URL: https://doi.org/10.25300/MISQ/2018/14558.

Shu, K., Cui, L., Wang, S., Lee, D., & Liu, H. (2019). DEFEND: Explainable fake news detection. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 395–405). URL: https://doi.org/10.1145/3292500.3330935.

Shu, K., Mahudeswaran, D., Wang, S., Lee, D., & Liu, H. (2020). FakeNewsNet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media. *Big Data*, 171–188, URL: https://doi.org/10.1089/big.2020.0062.

Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, *19*(1), 22–36, URL: https://doi.org/10.1145/3137597.3137600.

Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake news detection on social media: A data mining perspective. *SIGKDD Explorations Newsletter*, *19*(1), 22–36, URL: https://doi.org/10.1145/3137597.3137600.

Shu, K., Wang, S., & Liu, H. (2018). Understanding user profiles on social media for fake news detection. In *2018 IEEE conference on multimedia information processing and retrieval* (pp. 430–435). URL: https://doi.org/10.1109/MIPR.2018.00092.

Shu, K., Zhou, X., Wang, S., Zafarani, R., & Liu, H. (2019). The role of user profiles for fake news detection. In *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining* (pp. 436–439). URL: https://doi.org/10.1145/3341161.3342927.

Tao, T. (2012). *Topics in random matrix theory (vol. 132)*. American Mathematical Soc.

Van Der Maaten, L. (2014). Accelerating T-SNE using tree-based algorithms. *Journal of Machine Learning Research*, *15*(1), 3221–3245.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Proceedings of the 31st international conference on neural information processing systems* (pp. 6000—6010).

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., & Bengio, Y. (2018). Graph attention networks. In *6th international conference on learning representations*.

Vosoughi, S., Roy, D., & Aral, S. (2018). The spread of true and false news online. *Science*, 1146–1151, URL: https://doi.org/10.1126/science.aap9559.

Wang, Z., Chen, C., & Li, W. (2019). Information diffusion prediction with network regularized role-based user representation learning. *ACM Transactions on Knowledge Discovery from Data*, URL: https://doi.org/10.1145/3314106.

Wu, L., Rao, Y., Zhao, Y., Liang, H., & Nazir, A. (2020). DTCA: Decision tree-based co-attention networks for explainable claim verification. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 1024–1035). URL: https://doi.org/10.18653/v1/2020.acl-main.97.

Wu, K., Yang, S., & Zhu, K. Q. (2015). False rumors detection on sina weibo by propagation structures. In *2015 IEEE 31st international conference on data engineering* (pp. 651–662). URL: https://doi.org/10.1109/ICDE.2015.7113322.

Yang, X., Lyu, Y., Tian, T., Liu, Y., Liu, Y., & Zhang, X. (2020). Rumor detection on social media with graph structured adversarial learning. In *Proceedings of the twenty-ninth international joint conference on artificial intelligence* (pp. 1417–1423). URL: https://doi.org/10.24963/ijcai.2020/197.

Yu, K., Jiang, H., Li, T., Han, S., & Wu, X. (2020). Data fusion oriented graph convolution network model for rumor detection. *IEEE Transactions on Network and Service Management*, *17*(4), 2171–2181, URL: https://doi.org/10.1109/TNSM.2020.3033996.

Yu, F., Liu, Q., Wu, S., Wang, L., & Tan, T. (2017). A convolutional approach for misinformation identification. In *Proceedings of the twenty-sixth international joint conference on artificial intelligence* (pp. 3901–3907). URL: https://doi.org/10.24963/ijcai.2017/545.

Zarrinkalam, F., Fani, H., Bagheri, E., & Kahani, M. (2017). Predicting users' future interests on Twitter. In J. M. Jose, C. Hauff, I. S. Altıngovde, D. Song, D. Albakour, S. Watt, & J. Tait (Eds.), *Advances in information retrieval* (pp. 464–476).

Zarrinkalam, F., Kahani, M., & Bagheri, E. (2018). Mining user interests over active topics on social networks. *Information Processing & Management*, *54*(2), 339–357, URL: https://doi.org/10.1016/j.ipm.2017.12.003.

Zhang, J., Dong, Y., Wang, Y., Tang, J., & Ding, M. (2019). ProNE: Fast and scalable network representation learning. In *Proceedings of the twenty-eighth international joint conference on artificial intelligence* (pp. 4278–4284). URL: https://doi.org/10.24963/ijcai.2019/594.

Zhang, X., & Ghorbani, A. A. (2020). An overview of online fake news: Characterization, detection, and discussion. *Information Processing & Management*, *57*(2), Article 102025, URL: https://doi.org/10.1016/j.ipm.2019.03.004.

Zhao, T., Liu, Y., Neves, L., Woodford, O., Jiang, M., & Shah, N. (2020). Data augmentation for graph neural networks. arXiv preprint arXiv:2006.06830.

Zhao, Z., Resnick, P., & Mei, Q. (2015). Enquiring minds: Early detection of rumors in social media from enquiry posts. In *Proceedings of the 24th international conference on world wide web* (pp. 1395–1405). URL: https://doi.org/10.1145/2736277.2741637.

Zhou, X., & Zafarani, R. (2020). A survey of fake news: Fundamental theories, detection methods, and opportunities. *ACM Computing Surveys*, *53*(5), URL: https://doi.org/10.1145/3395046.