Neurocomputing 531 (2023) 114–124

Contents lists available at ScienceDirect

Neurocomputing

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

TSNN: A Topic and Structure Aware Neural Network for Rumor Detection

Zhuomin Chen [a], Li Wang [a,*], Xiaofei Zhu [b], Stefan Dietze [c,d]

[a] College of Data Science, Taiyuan University of Technology, Shanxi, Jinzhong 030600, China
[b] College of Computer Science and Engineering, Chongqing University of Technology, Chongqing 400054, China
[c] Knowledge Technologies for the Social Sciences, Leibniz Institute for the Social Sciences, Cologne 50667, Germany
[d] Institute of Computer Science, Heinrich-Heine-University Düsseldorf, Düsseldorf 40225, Germany

ARTICLE INFO

Article history:
Received 21 April 2022
Revised 16 December 2022
Accepted 11 February 2023
Available online 17 February 2023
Communicated by Zidong Wang

Keywords:
Rumor detection
Neural topic models
Topic credibility
Multi-task learning

ABSTRACT

Detecting rumors on social media and preventing its spread play a critical role for politics, economy, etc. Conventional studies mainly focus on exploiting the content or context of the source post, while they always ignore the rich topic information within the source post. To tackle this issue, in this paper, we propose a Topic and Structure Aware Neural Network (TSNN) for rumor detection. To be specific, we explore two kinds of topic signals, including a coarse-grained topic signal (i.e., topic credibility) and a fine-grained topic signal (i.e., latent topic representation), and tailor them to the task of rumor detection. Moreover, we introduce a new auxiliary task, i.e., topic credibility prediction, in order to effectively leverage the rich topic information within source posts. Finally, we develop a multi-task learning strategy that helps improve rumor detection performance by jointly learning the task of topic credibility prediction and user credibility prediction. Extensive experiments on three real-world datasets demonstrate that the proposed approach TSNN is superior to the state-of-the-art baseline methods.

1. Introduction

Social media provides a convenient platform for people to obtain information, express opinions, and communicate with each other. However, it also enables widespread disinformation with malicious intent, named rumors, at a high rate, causing a crisis of social confidence. Accordingly, detecting rumors is critical for maintaining a trustful circumstance on social media. In previous work, many research efforts have been devoted to identifying rumors by extracting textual features and adopting machine learning techniques, such as Support vector Machine (SVM) [1], Random Forest [2], and Decision Tree [3]. These methods can identify rumors to some extent. However, they primarily rely on feature engineering, which is usually data-dependent and can not cope with the new emerging false information.

Recently, deep learning techniques [4,5] have been widely used for rumor detection, which did not rely on feature engineering and achieved some good performance. They usually utilize the post's content, transfer patterns, user profiles, comments, sentiments, or other information to detect rumors. For example, Ma et al.[6] proposed a model RvNN that tracks the post propagation process and utilizes RNN with a self-attention mechanism to learn selectively temporal hidden representations of sequential posts for identifying rumors. A text-based model SemSeq4FD [7] was proposed to consider the global semantic relations feature, local sequential order feature, and the global sequential order feature among sentences simultaneously to detect fake news. Wang et al. [8] introduced Elementary Discourse Unit (EDU) as a suitable unit to improve text representation for fake news detection that includes the sequence-based EDU representations and the graph-based EDU representations.

Obviously, the text reflects that the content of the post is a major component of all published works, taking into account words, sentences, EDU, and text structure. However, the topic, as an important signal in a post, was all but ignored. In this paper, we will focus on the effect of topics on improving rumor detection. It can be observed that the credibility of posts usually has a strong correlation with their topics. We can observe that, for example, a source post related to a sensitive political topic may be a rumor as it has the potential to influence politics and the public. It can bring a great number of political profits, such as affecting the presidential election results (e.g., "Breaking: Two Explosions in the White House and Barack Obama Is Injured."). On the other hand, if a source post is related to technology, such as "Apple Just Invented the Pencil", it would be more prone to true information than those with sensitive topics. Fig. 1 shows the credibility of dif-

* Corresponding author.
E-mail addresses: chenzhuomin1266@link.tyut.edu.cn (Z. Chen), wangli@tyut.edu.cn (L. Wang), zxf@cqut.edu.cn (X. Zhu), stefan.dietze@gesis.org (S. Dietze).

https://doi.org/10.1016/j.neucom.2023.02.016
0925-2312/

(a) Weibo (b) Twitter15

Fig. 1. Examples of topic credibility learned on Weibo and Twitter15 datasets, respectively. LDA is used to assign each source post with its most salient topic, and the credibility of each topic is annotated based on the ratio of non-rumors for that topic in the training set. The number of topics for LDA is set to 50, and 10 of the 50 topics are randomly selected for visualization.

ferent topics in the Weibo [9] and Twitter15 [10] datasets. We can observe that the credibility of topics varies.

This indicates that the credibility of topics can be leveraged as an important clue for guiding the learning process of rumor detection. Some works also noticed the effect of the topic and proposed methods to detect fake news. Jin et al. [11] collected the posts related to one news event, obtained their topics and opinions, and exploited their conflicts to detect fake news events. Hu et al. [12] proposed to enhance the news representation by a heterogeneous graph of news text, topic, and entity representation from an external knowledge base for fake news detection. Our work is to identify the credibility of each source post, which is different from Jin's work, that is, to identify the news event, not the posts. An external knowledge base would help bring auxiliary information but also take more cost on computing. So, different from Hu's work, in this paper, we propose a new method that is just based on source posts and their contextual information on the Web without any external knowledge. We introduce the credibility of topics and develop a multi-task learning strategy to detect rumors. We use Text-CNN to learn the source post representations that capture the phrase-level features [13,14], and propose jointly modeling the latent topic of source posts as a weakly supervised signal to guide rumor detection.

However, obtaining the credibility of topics is not a trivial task since the corresponding topic information of each source post is not available. To handle this issue, we employ an unsupervised probabilistic topic model (LDA [15]) to extract the topics for each source post. We assign the topic with the highest probability value in the topic distribution of each source post as its corresponding topic. In addition, Jin et al.[11] and Hu et al.[12] only used LDA to extract the keywords of the original post and establish a relationship with the text, and LDA cannot be combined with the neural network for end-to-end training. Different from these two methods, taking advantage of the recent advance of neural topic models (NTM) [16,17], we employ NTM to learn source posts' latent topic representations as it enables end-to-end training of latent topic modeling and source post classification. It is worth noting that NTM provides a kind of 'high-level syntactic features' compared to the representations learned by CNN. Thus, they are complementary to each other, a topic comparison network is employed to fuse the two different kinds of syntactic features, i.e., phrase-level features and topic-level features.

Motivated by this, we propose a Topic and Structure Aware Neural Network (TSNN) for rumor detection, which not only enhances the representation learning of the source post by topic representation but also participates in model training with topic credibility. More specifically, our model mainly consists of two components, a *topic-aware text encoder* module and a *structure-aware user encoder* module. In *topic-aware text encoder*, we propose to exploit the topic signals in the training process of rumor detection. We incorporate two kinds of topic signals: a coarse-grained topic signal (i.e., topic credibility) and a fine-grained topic signal (i.e., topic representation). The former is developed to provide the credibility of topics as a weakly supervised clue to guide the learning process of source post representation. The latter is leveraged to better capture the semantics embedded in the source post. In *structure-aware user encoder*, the representations of publishers and communicators are generated from information disseminated by the source post. The topic-aware text representations and credibility-aware user representations are used to train a classifier to detect rumors. Fig. 2 illustrates the overall architecture of TSNN.

We carry out extensive experiments on three widely used datasets. The results show that our proposed approach TSNN is superior to these state-of-the-art rumor detection baselines on all three datasets. The main contributions of this paper are as follows:

- We propose to explore two kinds of topic signals, including a coarse-grained topic signal (i.e., topic credibility) and a fine-grained topic signal (i.e., latent topic representation), and tailor them to the task of rumor detection.
- We propose a multi-task learning strategy for rumor detection and jointly train the model on both topic credibility prediction and user credibility prediction.
- Extensive experiments are conducted on three benchmark datasets (i.e., Twitter15, Twitter16, and Weibo). The experimental results show that our proposed model considerably outperforms current state-of-the-art baseline methods in rumor detection.

The rest of the paper is organized as follows. In Section 2, we give a brief review of the related work. We introduce our proposed topic and structure aware neural network in Section 3. Section 4 discusses the experimental results of our empirical studies. In Section 5, we conclude the paper.

2. Related Work

Rumor detection is an important task in natural language processing and has recently attracted increasing attention due to its impact on public trust [18]. The early research works on rumor detection mainly focus on extracting textual features from the source post and applying traditional learning techniques (e.g., SVM [1], Random Forest [2], and Decision Tree[3]) to detect rumors. The textual features can be roughly grouped into two categories, i.e., low-level text features [19,20] and high-level text fea-

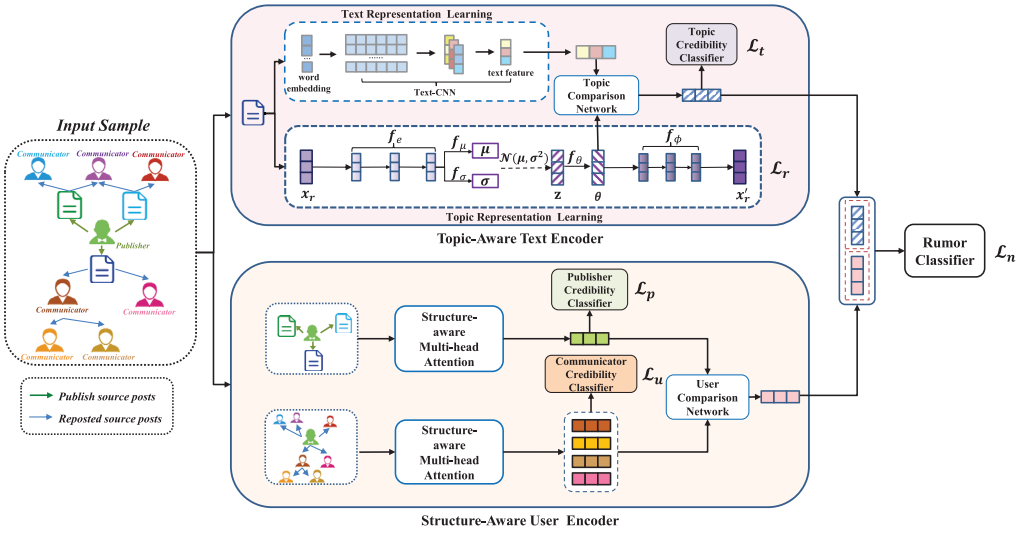

Fig. 2. The framework of the proposed approach TSNN.

tures [21,22]. The former attempts to extract features such as n-gram [19] or bag-of-words [20]. While the latter usually relies on extracting more complex features, e.g., sentiment features [21] or writing style consistency [22].

In recent years, due to the tremendous success of deep neural networks in various fields [23,24], many research efforts have been devoted to building deep neural network models for the task of rumor detection. Vaibhav et al. [25] proposed a graph neural network model for rumor detection, which models the semantic relationship between all sentence pairs in the source post for rumor detection. To deal with this issue, Yu et al. [26] extract key features of both misinformation and truth information scattered among an input sequence based on operations such as convolutional and k-max pooling. In addition, high-level interactions among significant features are also captured. As the importance of textural features may vary along time, Chen et al. [27] adopt soft-attention mechanism to aggregate distinct textual features with specific focus and capture contextual variations of relevant source posts over time.

In addition to utilizing textual content, many researchers have noticed that social context information is important for distinguishing rumors. More recently, some works further utilize the social context, such as diffusion structure, to assist the detection of rumors. Ma et al. [6] constructed a bottom-up and top-down tree-structured neural network (RvNN) for rumor detection. They leverage the recursive feature learning process along the tree structure to jointly model the source post contents and the responsive relationship among them.

Liu et al. [28] consider the diffusion path of source post story as a multivariate time series and build a time series classifier to capture both global and local variations of user characteristics along the diffusion path. Yuan et al. [13] capture the structural information by modeling global relationships among tweets, retweets, and users, and a novel global–local attention network is then proposed for rumor detection. Yuan et al. [14] propose a structure-aware multi-attention network (SMAN), which considers rumor detection as a multi-task classification task. Different from previous work, they explicitly adopt the credibility of users as weakly supervised information, and jointly optimize the rumor detection and user credibility prediction. As the diffusion structure usually contains unreliable relations, it would lead to inferior performance for the detection of rumors. To alleviate this issue, Wei et al. [29] propose to explore the uncertainty of the propagation structure. They present a Bayesian based model EBGCN, that replaces the edge weights of propagation graph by introducing the prior belief of the observed graph in order to control the message-passing. Moreover, they develop an edge-wise consistency training framework to maintain the consistency between the latent distributions of edges and node features. It can be seen from the above works that text content is essential information for rumor detection, and social context can also play an auxiliary role in rumor detection.

Different from these previous works, we argue that topic signals within source posts have a strong correlation with credibility, and it is beneficial to incorporate them to facilitate the rumor detection. To be specific, we explore the topic signals within source posts in two different ways, including a coarse-grained topic signal and a fine-grained topic signal.

3. Our Approach

In this section, we detail our proposed model Topic and Structure Aware Neural Network (TSNN) for rumor detection.

3.1. Problem Definition

This paper focuses on the rumor detection task. We denote the set of source posts as $N = (m_1, m_2, \cdots, m_{|N|})$, the set of publishers as $P = (p_1, p_2, \cdots, p_{|P|})$ and the set of communicators as $U = (r_1, r_2, \cdots, r_{|U|})$, where $|P|, |N|$ and $|U|$ are the number of publishers, source posts, and communicators, respectively. Each source post m_i has one publisher $p \in P$ and k communicators $\{r_j \in U\}_{j=1}^{k}$ who retweet the source post. Y represents the identification set of source posts. Our goal is to learn a classifier f for source posts to identify their credibility, that is, $f : \{N, P, U\} \rightarrow Y$.

3.2. Overview

The overall structure of the proposed model is shown in Fig. 2, which consists of two major components. The two components are Topic-Aware Text Encoder and Structure-Aware User Encoder.

(1)*Topic-Aware Text Encoder*. This encoder is designed to enrich source post representation by incorporating the topic signal (see Section 3.3). More precisely, two kinds of topic signals of

source posts have been exploited in the encoder, including the fine-grained topic signal (i.e., topic distribution) as well as the coarse-grained topic signal (i.e., topic credibility). The former is used to better understand the source post content by further exploring its underlying topic distribution, while the latter is leveraged to incorporate the credibility of topics as a weakly supervised signal to guide the learning process of source post representation.

(2)*Structure-Aware User Encoder*. It is developed to capture the representation of publishers and communicators based on the source post diffusion information (see Section 3.4). Given the source post propagation structure, we obtain the representation of publishers and communicators respectively by utilizing their corresponding credibility as weakly supervised information. A user comparison network is then leveraged to fuse the two representations and obtain the structure-aware user representation. Finally, the topic-aware text representation and the structure-aware user representation are combined and fed into a classification layer.

3.3. Topic-Aware Text Encoder

The *topic-aware text encoder* is comprised of four sub-modules, i.e., text representation learning, topic representation learning, topic comparison network, and topic credibility classifier. Text representation learning and topic representation learning are designed to obtain the semantic representation and topic representation of source post, respectively. Then the two representations are fed into a topic comparison network to obtain a fused text representation. At last, a topic credibility classifier is employed to incorporate the credibility of topics as a weakly supervised signal to guide the learning process of source post representation.

3.3.1. Text Representation Learning

Let $m_i = (w_1, w_2, \ldots, w_L)$ be the i-th source post which consists of L words, we embed each word in the source post into a low-dimensional real-valued vector with embedding matrix $\mathbf{E} \in \mathbb{R}^{|V| \times d}$, where $|V|$ is the vocabulary size and d is the dimensionality of word embeddings. With the word embeddings of the source post, we obtain $\mathbf{X}^i = (\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_L)$ with $\mathbf{x}_j \in \mathbb{R}^d$ be a d-dimensional word embedding corresponding to the j-th word in the source post. Let $\mathbf{x}_{j:j+k}$ denote the concatenation of the word embedding sequence from the j-th word to the $(j+k)$-th word, i.e., $\mathbf{x}_{j:j+k} = \mathbf{x}_j \oplus \mathbf{x}_{j+1} \oplus \cdots \oplus \mathbf{x}_{j+k}$.

After that, a Convolutional Neural Network (CNN) [30] is leveraged on top of the word embeddings $\mathbf{X}^i$ to produce the semantic representation of the source post. To be specific, a convolution operation involves a filter $\mathbf{w} \in \mathbb{R}^{hd}$, which is applied to a window of h words to produce a new feature. A feature c_j is generated from a window of words $\mathbf{x}_{j:j+h-1}$ as follows:

$$c_j = f(\mathbf{w} \cdot \mathbf{x}_{j:j+h-1} + b), \tag{1}$$

where f is an activation function and $b \in \mathbb{R}$ is a bias term. This filter is applied to each possible window of words in the source post $(\mathbf{x}_{1:h}, \mathbf{x}_{2:h+1}, \cdots, \mathbf{x}_{L-h+1:L})$ to produce a feature map $\mathbf{c} = (c_1, c_2, \cdots, c_{L-h+1})$ with $\mathbf{c} \in \mathbb{R}^{L-h+1}$. The dimensionality of the feature map generated by each filter will vary as a function of the filter region size. A max-pooling operation is applied to each feature map, obtaining the maximum value as the feature corresponding to this filter. The outputs generated from all filters will be concatenated to obtain the text representation $\mathbf{m}_i \in \mathbb{R}^{n_k}$ by employing n_k kernel.

3.3.2. Topic Representation Learning

We employ the neural topic models (NTM) [16,17] to learn topic representation of source posts. The principle of NTM is derived from the Variational Auto-Encoder (VAE) [31], which consists of an encoder and a decoder to simulate the reconstruction of source posts. Specifically, the i-th source post m_i is first represented by a bag-of-words vector $\mathbf{v}_i \in \mathbb{R}^V$, where V is the size of the vocabulary. Then an encoder is employed to convert $\mathbf{v}_i$ into a latent vector $\mathbf{z}_i \in \mathbb{R}^K$, where $\mathbf{z}_i$ represents the topic of the i-th source post, and K indicates the number of topics. The encoder is used to estimate prior variables μ and σ, which is used to infer the intermediate topic representation $\mathbf{z}_i$. Formally, we have

$$\mu = f_\mu(f_e(\mathbf{v}_i)), \tag{2}$$

$$log\sigma = f_\sigma(f_e(\mathbf{v}_i)), \tag{3}$$

where $f_\mu(\cdot), f_e(\cdot)$ and $f_\sigma(\cdot)$ are ReLU-activated neural perceptrons.

After that, a decoder conditioned on $\mathbf{z}_i$ is incorporated to reconstruct $\mathbf{v}_i$ and output a new BoW vector $\mathbf{v}_i\prime$. Each topic t is represented with a topic-word distribution ϕ_t over the vocabulary, and the source post m_i has a topic mixture represented by $\theta_i \in \mathbb{R}^K$, where θ_i is constructed by Gaussian softmax [16]. To simulate how source post m_i is generated, the decoder conducts the following steps:

- Draw latent topic variable $\mathbf{z}_i \sim N(\mu, \sigma^2)$
- Topic mixture $\theta_i = softmax(f_\theta(\mathbf{z}_i))$
- For each word $w \in \mathbf{v}_i$, Draw $w \sim softmax(f_\phi(\theta_i))$

where $f_\theta(\cdot)$ and $f_\phi(\cdot)$ are ReLU-activated neural perceptrons. At last, the topic mixture θ_i will serve as the topic representation of the i-th source post.

The loss function of NTM is defined as follows:

$$\mathscr{L}_r = \sum_{i=1}^{|N|} (D_{KL}(p(\mathbf{z}_i) \| q(\mathbf{z}_i|\mathbf{v}_i)) - \mathbb{E}_{q(\mathbf{z}_i|\mathbf{v}_i)}[p(\mathbf{v}_i|\mathbf{z}_i)]), \tag{4}$$

where $p(\mathbf{z}_i)$ represents the standard prior probability, $q(\mathbf{z}_i|\mathbf{v}_i)$ and $p(\mathbf{v}_i|\mathbf{z}_i)$ represent the results of encoder and decoder, respectively. The first term is the Kullback–Leibler (KL) divergence loss and the second term reflects the reconstruction loss.

3.3.3. Topic Comparison Network

For the i-th source post m_i, we obtain the CNN-based text representation $\mathbf{m}_i$ and topic-based text representation θ_i. Then we fuse both of them by a topic comparison network. We calculate the comparison vector $\mathbf{m}_i^{text}$ between $\mathbf{m}_i$ and θ_i as follows:

$$\mathbf{m}_i^{text} = f_c(\mathbf{m}_i, \mathbf{W}_\theta \cdot \theta_i), \tag{5}$$

where $f_c(\cdot)$ is the comparison function, $\mathbf{W}_\theta$ is the transformation matrix which transforms representations from the topic-level representation space to the phrase-level representation space. To maintain the representation closeness and relevance of $\mathbf{m}_i$ and θ_i, we design our topic comparison network as:

$$f_c(\mathbf{x}, \mathbf{y}) = [\mathbf{x}; \mathbf{y}; \mathbf{x} - \mathbf{y}; \mathbf{x} \odot \mathbf{y}]\mathbf{W}_c + \mathbf{b}_c, \tag{6}$$

where $\mathbf{W}_c \in \mathbb{R}^{4d \times d}$ is a transformation matrix and $\odot$ is hadamard product, i.e., element-wise product, and $\mathbf{b}_c \in \mathbb{R}^{1 \times d}$ is the bias vector.

3.3.4. Topic Credibility Classifier

We propose to incorporate the credibility of topics as a weakly supervised signal to guide rumor detection. Since the topic information of each source post is not available, we leverage an unsupervised probabilistic topic model (LDA [15]) to extract the topics for each source post. Specifically, we treat each source post as a pseudo-document, and the generative process of LDA is formalized as follows:

$$\theta_m \sim Dir(\alpha_0), \quad \text{for } m \in N$$
$$z_n \sim Multi(\theta_m), \quad \text{for } n \in [1, n_m]$$
$$w_n \sim Multi(\beta_{z_n}), \quad \text{for } n \in [1, n_m]$$

, where N is the source post corpus, α_0 is the hyper-parameter of the Dirichlet prior, θ_m denotes the topic distribution of the source post m, n_m is the total number of words in source post m, and β_{z_n} represents the topic distribution over words given topic assignment z_n. We consider the topic with the highest probability value in the topic distribution θ_m as the topic of that source post m.

After we assign a topic to each source post, the credibility of each topic is annotated based on the ratio of non-rumors in that topic in the training set. Specifically, we define three levels of credit for topics ($c = \{0, 1, 2\}$): (1) "0" means "reliable", which means that the proportion of non-rumors and true rumors in the topic is greater than or equal to 75% (The "true rumor" denotes a source post that debunks a certain rumor [13].); (2) "2" means "unreliable", which means the proportion of false rumors and unverified rumors in the topic is greater than or equal to 75%. (3) "1" means "uncertain", indicating that the proportion of different types of source posts in this topic belongs to other situations than the above "0" and "2".

The text embedding that output from the comparison network is used to predict the topic credibility of the source post:

$$p_i(c) = \text{softmax}(\mathbf{m}_i^{text}\mathbf{W}_t + \mathbf{b}_t), \tag{7}$$

where $\mathbf{W}_t \in \mathbb{R}^{d \times |c|}$ is the transformation matrix, and $|c|$ is the number of distinct topic credibility.

Finally, the topic credibility prediction task can be transformed into a multi-classification task. The cross-entropy loss is applied as the optimization function:

$$\mathscr{L}_t = -\sum_{i=1}^{|N|} y_i^{(t)} \log p_i(c), \tag{8}$$

where $y_i^{(t)}$ is the ground-truth topic credibility of the i-th source post.

3.4. Structure-Aware User Encoder

The *structure-aware user encoder* is comprised of three submodules, i.e., publisher credibility classifier, communicator credibility classifier, and user comparison network. Publisher credibility classifier and communicator credibility classifier are developed to obtain effective representations of publishers and communicators, respectively. Then the two representations are fed into the user comparison network to obtain the structure-aware user representation.

3.4.1. Publisher Credibility Classifier

For each source post publisher p, we construct a heterogeneous graph $G_p = (V_p, E)$ where V_p consists of both publisher nodes and source post nodes, and $E_{ij} = 1$ indicates the i-th publisher posts the j-th source post. Let $\mathbf{A}^{pn} \in \mathbb{R}^{|P| \times |N|}$ be the corresponding adjacency matrix, $\mathbf{P} \in \mathbb{R}^{|P| \times d}$ and $\mathbf{N} \in \mathbb{R}^{|N| \times d}$ be the initialized representations of publishers and source posts respectively, where $|P|$ and $|N|$ denote the number of publishers and source posts. We input the heterogeneous graph G_p into an extended multi-head attention network and calculate publishers' embeddings as follows:

$$\mathbf{H}_l = \text{softmax}\left(\frac{\mathbf{P}\mathbf{W}_l\mathbf{N}^T}{\sqrt{d}} \odot (\mathbf{D}^{\mathbf{p}})^{-\frac{1}{2}}\mathbf{A}^{pn}(\mathbf{D}^{\mathbf{n}})^{-\frac{1}{2}}\right)\mathbf{N}, \tag{9}$$

where $\mathbf{D}^{\mathbf{p}}$ and $\mathbf{D}^{\mathbf{n}}$ are diagonal matrices with $\mathbf{D}^{\mathbf{p}}_{ii} = \sum_j \mathbf{A}^{pn}_{ij}$ and $\mathbf{D}^{\mathbf{n}}_{jj} = \sum_i \mathbf{A}^{pn}_{ij}$, $\mathbf{W}_l \in \mathbb{R}^{d \times d}$ is the trainable parameter of the l-th ($l \in [1, h]$) head, and h is the number of heads in the multi-headed attention. We concatenate the output of multi-head attention and convert them to the final output through a fully connected layer:

$$\mathbf{P}' = \text{ELU}([\mathbf{H}_1; \mathbf{H}_2; \ldots; \mathbf{H}_l]\mathbf{W}_\alpha) + \mathbf{P}, \tag{10}$$

where $[;]$ is the concatenation operator, $\mathbf{W}_\alpha \in \mathbb{R}^{hd \times d}$ is a linear transformation matrix, ELU is activation function. After that, we obtain the publisher representations $\mathbf{P}' = (\mathbf{p}'_1, \mathbf{p}'_2, \ldots, \mathbf{p}'_{|P|})$ where $\mathbf{p}'_i \in \mathbb{R}^d$ is the representation of the i-th publisher.

Finally, we use the publisher representations $\mathbf{P}'$ to predict the credibility of each publisher. Formally, we have:

$$p_i(c|G_p) = \text{softmax}(\mathbf{p}'_i\mathbf{W}_p + \mathbf{b}_p), \tag{11}$$

where $\mathbf{W}_p \in \mathbb{R}^{d \times |c|}$ is a transformation matrix, and $|c|$ is the number of levels of publishers' credibility scores. We use the cross-entropy loss as the objective function:

$$\mathscr{L}_p = -\sum_{i=1}^{|P|} y_i^{(p)} \log p_i(c|G_p), \tag{12}$$

where $y_i^{(p)}$ is the ground-truth credibility score of the i-th publisher.

3.4.2. Communicator Credibility Classifier

In Section 3.1, it is assumed that during the information propagation between publishers and communicators, each source post is forwarded by at most k communicators. Based on the propagation, we construct a heterogeneous graph $G_u = (V_u, E)$. Then we feed the initial representations of communicators $\mathbf{R} \in \mathbb{R}^{|U| \times d}$ and G_u into a similar multi-head attention network as in Section 3.4.1, and obtain the communicator representations $\mathbf{U} \in \mathbb{R}^{|U| \times k \times d}$, where $|U|$ denotes the number of communicators.

We predict the communicator credibility as follows:

$$p_{ij}(c|G_u) = \text{softmax}(\mathbf{u}_{ij}\mathbf{W}_u + \mathbf{b}_u), \tag{13}$$

where $\mathbf{u}_{ij} \in \mathbf{U}, i \in [1, 2, \ldots, |U|]$, and $j \in [1, 2, \ldots, k]$, $\mathbf{W}_u \in \mathbb{R}^{d \times |c|}$ is a transformation matrix, $\mathbf{b}_u \in \mathbb{R}^{|c|}$ is a bias vector. Similarly, we apply the cross-entropy loss as the objective function:

$$\mathscr{L}_u = -\sum_{i=1}^{|U|}\sum_{j=1}^{k} y_{ij}^{(u)} \log p_{ij}(c|G_u), \tag{14}$$

where $y_{ij}^{(u)}$ is the ground-truth credibility of communicator u_{ij}.

3.4.3. User Comparison Network

For the i-th source post m_i, we aggregate its corresponding publisher representation $\mathbf{p}'_i \in \mathbb{R}^d$ and communication representation $\mathbf{U}_i \in \mathbb{R}^{k \times d}$. To be specific, we first employ the attention mechanism to aggregate k communicators' representation for m_i. Formally, we have:

$$\mathbf{u}'_i = \sum_{j=1}^{k} \alpha_{ij}\mathbf{u}_{ij}, \tag{15}$$

$$\alpha_i = \text{softmax}\left(\mathbf{n}_i\mathbf{U}_i^T\right), \tag{16}$$

where $\mathbf{U}_i = (\mathbf{u}_{i1}, \mathbf{u}_{i2}, \ldots, \mathbf{u}_{ik})$, $\mathbf{u}_{ij} \in \mathbb{R}^d$, $\mathbf{n}_i \in \mathbb{R}^{1 \times d}$ is the representation of source post m_i in the initialized text embeddings $\mathbf{N} \in \mathbb{R}^{|N| \times d}$, and $\alpha_i = (\alpha_{i1}, \cdots, \alpha_{ik})$ are the attention weights.

Then, we merge the publisher representation $\mathbf{u}'_i$ and the communicator representations $\mathbf{p}'_i$ of source post m_i, and obtain the source post representation from the user perspective. Formally, we have:

$$\mathbf{m}_i^{user} = f_e(\mathbf{p}'_i, \mathbf{u}'_i). \tag{17}$$

Herein, we adopt the comparison network to merge the information, which is formulated as follows:

$$f_e(\mathbf{x},\mathbf{y}) = [\mathbf{x};\mathbf{y};\mathbf{x}\odot\mathbf{y};\mathbf{x}-\mathbf{y}]\mathbf{W}_e + \mathbf{b}_e, \tag{18}$$

where $\mathbf{W}_e \in \mathbb{R}^{4d\times d}$ and $\mathbf{b}_e \in \mathbb{R}^{d}$are trainable parameters.

3.5. Model Training

After we obtain the source post representations from the topic perspective (i.e., $\mathbf{m}_i^{text}$) and the user perspective (i.e., $\mathbf{m}_i^{user}$), we concatenate $\mathbf{m}_i^{text}$ and $\mathbf{m}_i^{user}$ and feed the concatenated representation into a fully connected layer. Then we adopt a softmax layer to calculate the probability distribution of the source post m_i. Formally, we have:

$$p(m_i) = \text{softmax}\big([\mathbf{m}_i^{text};\mathbf{m}_i^{user}]\mathbf{W}_m + \mathbf{b}_m\big), \tag{19}$$

where $\mathbf{W}_m \in \mathbb{R}^{4d\times|y|}$ is a transformation matrix and $\mathbf{b}_m \in \mathbb{R}^{|y|}$is bias vector.$|y|$ is the number of types of source post labels. For the rumor detection, we utilize the cross-entropy loss as the objective function:

$$\mathscr{L}_n = -\sum_{i=1}^{|N|} y_i^{(n)} \log p(m_i), \tag{20}$$

Where $y_i^{(n)} \in Y$ is the ground-truth label of source post m_i.

Finally, we use the linear combination to define the training objective of the entire model:

$$\mathscr{L} = \beta_p\mathscr{L}_p + \beta_u\mathscr{L}_u + \beta_r\mathscr{L}_r + \beta_t\mathscr{L}_t + \beta_n\mathscr{L}_n, \tag{21}$$

where $\beta_p, \beta_u, \beta_r, \beta_t$ and β_n are model hyper-parameters. Under the guidance of labeled data in training set, our model is trained via backpropagation.

4. Experiments

4.1. Datasets

To evaluate the performance of our model, we use three real-world datasets, namely Twitter15 [10], Twitter16 [10] and Weibo [9]. The former two datasets are derived from Twitter, and the third dataset is collected from a popular social media website in China. Table 1 shows the statistics of the datasets. Both Twitter15 and Twitter16 have four classes, namely Non-rumor (NR), False Rumor (FR), Unverified Rumor (UR), and True Rumor (TR). Compared with Twitter15 and Twitter16, the classes of Weibo are more coarse-grained. It contains two classes, i.e., Non-rumor (NR) and False Rumor (FR), which predict whether the source post is trustworthy or not. Following [14], we first randomly select 10% of data as the validation set, and then split the remaining data into training and test sets at a ratio of 3:1.

4.2. Baselines and Metrics

We compare our TSNN with twelve state-of-the-art baseline methods for the task of rumor detection. These baseline methods can be grouped into two categories, i.e., feature based methods and deep learning based methods.

(1) Feature based methods:

- DTC [32]: This is a decision tree model based on supervised learning, which extracts relevant features from each annotated topic to build a classifier, automatically determines whether a topic corresponds to valuable information and evaluates source post authenticity.
- SVM-RBF [3]: This model trains a Support Vector Machine (SVM) classifier with a Radial Basis Function (RBF) kernel function to identify rumors using features based on content, account, and propagation, respectively.
- SVM-TS [33]: This is a time-series model based on the rumors' life cycle, which utilizes time-series modeling techniques to capture broad social contextual information.
- DTR [34]: DTR is a method based on user query phrases. It aims to cluster tweets containing enquiry patterns, and collect related tweets without the simple phrases. Then it ranks the clusters based on statistical features based on properties of the signal tweets within the cluster.
- RFC [35]: This method combines user, structure, language, and temporal features to study the cumulative propagation pattern of rumors over time, tracking changes in the predictive ability of rumor features.
- cPTK [10]: It uses a classifier with a propagation tree kernel that learns to identify discriminative cues for rumors at a fine-grained level by evaluating the similarity between propagation tree structures.

(2) Deep Learning based methods

- GRU [9]: A RNN-based model that models social contextual information of events as a variable time sequence, learns temporal and textual representations of rumors.
- RvNN [6]: This recurrent neural network deeply integrates structural and content semantics and utilizes bottom-up and top-down tree structures for rumor detection.
- PPC [28]: This model combines a time series classifier with recurrent and convolutional networks to analyze the changes in user characteristics along the propagation path.
- GLAN [13]: The model combines local semantic and global structural information for rumor detection, and takes all source post content, comments and user interactions as global relations to form a heterogeneous graph.
- EBGCN [29]: This model adaptively adjusts the uncertainty of latent relationships in the propagating structure through a Bayesian approach and uses an edge-consistency training framework to enhance the consistency of latent relationships, providing structural features for rumor detection.
- SMAN [14]: This method combines source post content, posting, and forwarding relationships of publishers and communicators, treats publisher and communicator credibility as weakly supervised information, and jointly optimizes rumor detection and user credibility prediction.

To evaluate the performance produced by all comparing methods, we use the accuracy (Acc) as the overall evaluation metric for all three datasets. To evaluate the model performance for each class,

Table 1
Statistics of the datasets.

	# source posts	# NR	# FR	# UR	# TR	# users	# retweets
Twitter15	1490	374	370	374	372	276,663	331,612
Twitter16	818	205	205	203	205	173,487	204,820
Weibo	4664	2351	2313	0	0	2,746,818	3,805,656

we leverage precision (Pre), recall (Rec), and F1 score (F1) as metrics for the dataset Weibo, while adopt the F1 score (F1) as the evaluation metric for the other two datasets, i.e., Twitter15 and Twitter16.

4.3. Experimental Settings.

For model training, we employ the Adam algorithm [36] to update our model parameters, and set the initial learning rate as $\{1.7e-3, 2e-3, 0.9e-3\}$ for Twitter15, Twitter16, and Weibo, respectively. For neural topic model, we set the number of topics K to 50. The word embeddings are randomly initialized with the embedding size of 300. The convolution size of CNN in the text representation learning module is set to (3,4,5) with each size corresponding to 100 kernels. We set the number of heads in the structure-aware multi-head attention for the three datasets Twitter15, Twitter16 and Weibo as $\{10, 8, 7\}$, respectively. The parameters $\beta_p, \beta_u, \beta_r, \beta_t$, and β_n in Eq. (21) are empirically set to $\{1, 1, 1, 0.1, 1\}$, respectively.

4.4. Results and analysis

Table 2 shows the performance comparison of our model TSNN and all baseline methods on Twitter15. From the results, we can observe that TSNN is superior to all comparing methods in term of accuracy. Compared to the two best performing baselines, i.e., EBGCN and SMAN, our TSNN achieves the overall performance improvements of 3.1% and 0.7% respectively in terms of accuracy. This verifies the effectiveness of our proposed approach TSNN which further explores both fine-grained and coarse-grained topic signals. The fine-grained topic signal is leveraged to capture the underlying topic distribution, and the coarse-grained topic signal is employed to model the credibility of topics.

In Table 2, we also show the F1 score of the proposed approach and all baselines with respect to each of the four classes (i.e., NR, FR, TR, UR). We can see that on most of the classes, such as NR, TR, and UR, our proposed model TSNN consistently outperforms all comparing methods. While for the UR class (i.e., unverified rumors), TSNN demonstrates a better performance than all baseline methods except EBGCN. This may be attributed to that the UR class is more ambiguous as compared to other three classes. EBGCN obtains a better performance since it can effectively handle this issue by adaptively controlling the message-passing based on the prior belief.

Table 3 shows the performance of all comparing methods on Twitter16. Similar to the results on Twitter15, our proposed method demonstrates a better overall performance than the two best comparing baselines in terms of Acc, e.g., the performance improvements over EBGCN and SMAN are 3.4% and 1.8%, respectively. In addition, the F1 scores on the four classes of Twitter16 are consistent to that of Twitter15. The reason is that the two datasets are all collected from the same platform and the main difference is that they have different data sizes. Therefore, our proposed method TSNN is prone to have similar performance on the two datasets.

Table 4 illustrates the performance of all comparing methods on Weibo. We cam observe that TSNN shows a superior overall performance compared to all baselines in terms of Acc, e.g., the performance improvements over the best performing baseline SMAN is 0.4%. Note that there are only two classes, i.e., Non-rumor (NR) and False Rumor (FR). From Table 4, we can see that TSNN is consistently superior to all baselines.

4.5. Ablation Study

In this section, we perform an ablation study to analyze the role of each component in TSNN. In particular, we have the following variants:

- **Users Only**: We only apply the *structure-aware user encoder* module in TSNN to model the information of publishers and communicators of source posts for classification.

Table 2
Performance comparison of our TSNN and all state-of-the-art baseline methods on Twitter15. We highlight the two best performing methods (bold: best result, underlined: second-best result). Results marked ∗ taken from [13] and ◆ taken from [29].

Model	Acc	NR	FR	TR	UR
		F1	F1	F1	F1
DTR∗	40.9	50.1	31.1	36.4	47.3
DTC∗	45.4	73.3	35.5	31.7	41.5
RFC∗	56.5	81.0	42.2	40.1	54.3
SVM-RBF∗	31.8	45.5	3.7	21.8	22.5
SVM-TS∗	54.4	79.6	47.2	40.4	48.3
cPTK∗	75.0	80.4	69.8	76.5	73.3
GRU∗	64.6	79.2	57.4	60.8	59.2
RvNN∗	72.3	68.2	75.8	82.1	65.4
PPC∗	84.2	81.1	87.5	81.8	79.0
GLAN∗	90.5	92.4	91.7	85.2	92.7
EBGCN◆	89.2	86.9	89.7	**93.4**	86.7
SMAN	91.4	90.6	91.1	86.2	92.8
TSNN	**92.0**	**92.8**	**91.9**	89.3	**94.0**

Table 3
Performance comparison of our TSNN and all state-of-the-art baseline methods on Twitter16. We highlight the two best performing methods (bold: best result, underlined: second-best result). Results marked ∗ taken from [13] and ◆ taken from [29].

Model	Acc	NR	FR	TR	UR
		F1	F1	F1	F1
DTR∗	41.4	39.4	27.3	63.0	34.4
DTC∗	46.5	64.3	39.3	41.9	40.3
RFC∗	58.5	75.2	41.5	54.7	56.3
SVM-RBF∗	32.1	42.3	8.5	41.9	3.7
SVM-TS∗	57.4	75.5	42.0	57.1	52.6
cPTK∗	73.2	74.0	70.9	83.6	68.6
GRU∗	63.3	77.2	48.9	68.6	59.3
RvNN∗	73.7	66.2	74.3	83.5	70.8
PPC∗	86.3	82.0	89.8	84.3	83.7
GLAN∗	90.2	92.1	86.9	84.7	96.8
EBGCN◆	91.5	87.9	90.6	**94.7**	91.0
SMAN	92.9	93.6	90.5	90.5	96.8
TSNN	**94.6**	**93.8**	**93.0**	93.5	**97.9**

Table 4
Performance comparison of our TSNN and all state-of-the-art baseline methods on Weibo. We highlight the two best performing methods (bold: best result, underlined: second-best result). Results marked ∗ taken from [13].

Model	Acc	NR			FR		
		Pre	Rec	F1	Pre	Rec	F1
DTR∗	73.2	72.6	74.9	73.7	73.8	71.5	72.6
DTC∗	83.1	81.5	84.7	83.0	84.7	81.5	83.1
RFC∗	84.9	94.7	73.9	83.0	78.6	95.9	86.4
SVM-RBF∗	81.8	81.5	82.4	81.9	82.2	81.2	81.7
SVM-TS∗	85.7	87.8	83.0	85.7	83.9	88.5	86.1
GRU∗	91.0	95.2	86.4	90.6	87.6	95.6	91.4
PPC∗	92.1	94.9	88.9	91.8	89.6	96.2	92.3
GLAN∗	94.6	94.9	94.3	94.6	94.3	94.8	94.5
EBGCN	90.2	91.2	88.8	89.5	87.1	90.5	88.2
SMAN	95.1	95.6	94.7	95.2	94.7	95.6	95.1
TSNN	**95.5**	**96.2**	**94.9**	**95.5**	**94.9**	**96.2**	**95.5**

- **Text Only**: It only uses *text representation learning* module to model the textual information of original posts for rumor detection.
- **Users + Text**: This variant leverages the *structure-aware user encoder* module as well as the *text representation learning* module to learn source post representations for detection. It is worth noting that this variant is equivalent to the baseline method SMAN [14].
- **Users + Topic**: Different to Users + Text, this variant replaces the *text representation learning* module with the topic distribution of source posts learnt by NTM [16].
- **Users + Text + Topic**: This variant is a combination of the above two variants, i.e., Users + Text and Users + Topic. To obtain a better representation of the source post, it aggregates information from the *structure-aware user encoder* module, the *text representation learning* module, and the topic distribution of source posts learnt by NTM.
- **Users + Text + Topic Credibility**: This is a variant that extends the variant Users + Text by further incorporating the topic credibility as a weakly supervised information to guide the representation learning process of source posts. It can also be considered as an improved variant of SMAN by introducing an auxiliary task, i.e., topic credibility classification.
- **Users + Topic + Topic Credibility**: Similarly, based on the variant Users + Topic, we further introduce the topic credibility as a weakly supervised information in order to learn better source post representations.
- **Users + Text + Topic + Topic Credibility**: This is our proposed method TSNN, which learns rumor detection by capturing the source post content, latent topic representations, as well as exploring the credibility of publishers, communicators, and topics.

The results of the ablation study on all datasets are reported in Table 5. From Table 5, we can have the following observations:

- The variant 'Users Only', which utilizes only the information of publishers and communicators, obtains the lowest accuracy. The accuracy is greatly improved when we improve 'Users Only' by introducing the latent topic information (i.e., 'Users + Topic'). When we further take the 'Topic Credibility' into consideration (i.e., 'Users + Topic + Topic Credibility'), the performance will be improved again.
- Based on "Text Only", the performance of the variant "Users + Text" obtained after adding publishers and communicators has been improved. This shows that the structure-aware user encoder module plays an active role in the model.
- Among all variants, the performance of these text-based variants (such as 'Users + Text', 'Users + Text + Topic', 'Users + Text + Topic Credibility) are significantly better than that of the remaining three text-free variants (i.e., 'Users Only', 'Users + T opic', 'Users + Topic + Topic Credibility'). For example, on the Twitter15 dataset, adding text information to the variants 'Users Only' and 'Users + Topic' will lead to performance increasement of 76.4% and 37.5% respectively. Similar trends can be observed on the other two datasets. This demonstrates that the textual information within the source post contains a critical signal for detecting rumors, and incorporating this information can considerably boost the performance.
- Compared to the TSNN model (i.e., 'Users + Text + Topic + Topic Credibility'), all variants with components removed show a significant drop in performance, suggesting that each component plays a positive role. This is because our proposed approach can effectively model all these key components in a proper way.

Table 5
Ablation study on all datasets removing different parts of our proposed model TSNN.

Variants				Twitter15	Twitter16	Weibo
Users	Text	Topic	Topic Credibility	Accuracy	Accuracy	Accuracy
✓				51.8	52.2	89.9
	✓			83.2	82.6	93.2
✓	✓			91.4	92.9	95.1
✓		✓		66.7	65.8	93.3
✓	✓	✓		91.7	94.0	95.2
✓	✓		✓	91.1	93.5	94.9
✓		✓	✓	68.4	66.8	93.9
✓	✓	✓	✓	**92.0**	**94.6**	**95.5**

4.6. Impact of the number of topics

In this section, we investigate how the number of topics affects the model's performance. We vary the number of topics in $\{25, 50, 75, 100, 125, 150\}$. Fig. 3 shows the performance of TSNN with the different number of topics on all three datasets. We can observe that the number of topics K affects the performance of TSNN considerably. On the Twitter16 dataset, the performance of TSNN keeps increasing with the number of topics growing and achieves the highest accuracy when the number of topics equals 50. When we continue to increase K, the performance will drop gradually. One possible reason is that when the number of topics becomes too large, the number of source posts assigned to each topic will be small what inevitably leads to improper estimation of topic credibility. On the Weibo dataset, the performance of TSNN also rises first and reaches the peak when $K = 50$. When we continue to raise the number of topics, the performance will drop slowly as compared to that on the Twitter16 dataset. This is because the size of the Weibo dataset is larger than the Twitter16 dataset. When we increase the number of topics, there are still enough source posts assigned to each topic for estimating proper topic credibility. Similar trend is observed on the Twitter15 dataset.

4.7. Parameter Sensitivity Analysis

In this section, we analyze the influence of the parameters $\beta_p, \beta_u, \beta_r, \beta_t$ and β_n which are used to balance the contribution of different sub-tasks in the objective function (see Eq. 21), including publisher credibility classification, communicator credibility classification, neural topic model reconstruction, topic credibility classification and rumor classifier in the objective function, respectively. For parameters β_p, β_u and β_r, we vary each of them from 0 to 1.2 with an interval 0.2. For the parameter β_t, we vary it in {0, 0.0001, 0.001, 0.01, 0.1, 1, 10}. For the parameter β_n, we vary it from 0.2 to 1.2 with an interval 0.2. To study the influence

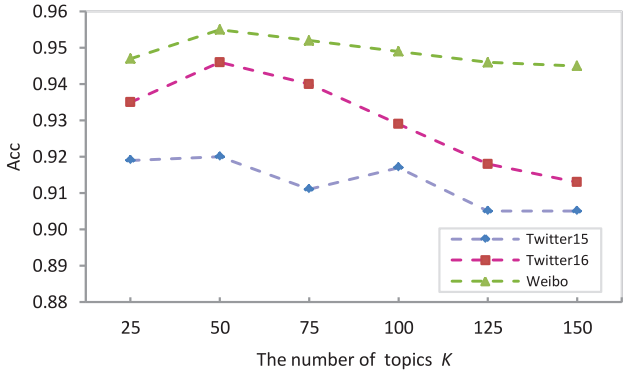

Fig. 3. Impact of the number of topics.

of each individual parameter on the classification results, we refer to the previous method [37], only changing the value of a specific parameter and fixing the remaining parameters to their respective optimal values.

Fig. 4(a) shows the performance of the proposed model with respect to parameter β_p. We can see that the performance of our model continues to rise when we increase β_p and reaches the peak when β_p = 1.0. If we further increase β_p, it starts to decrease. Similar results are observed Fig. 4(b) for the parameter β_u. The results indicate that incorporating the credibility of publisher and communicator plays a vital role for assisting the task rumor detection. Fig. 4(c) demonstrates the influence of employing NTM to model source posts' latent topic representations. We can observe that the performance of our method TSNN increases gradually when we raise β_r, and it achieves the best performance when $\beta_r = 1.0$. If we further increase β_r, the model performance starts to drop. The results verify that modeling source posts' latent topic representations with NTM is critical for affecting the performance of our proposed method. Fig. 4(d) presents the impact of introduce the topic credibility in our method, which is leveraged as a weak supervised information to guide the process of model training. With increase of β_t, we can observe a gradual performance improvement. The best performance is obtained when $\beta_t = 0.1$, which is followed by a quick performance drop. It indicates that introducing the topic credibility impacts on the performance of the proposed method. Fig. 4(e) shows the influence of the rumor detection, which is the main task of our proposed method. Not surprisingly, the proposed model is very sensitive to β_n and achieves the best performance when $\beta_n = 1.0$.

4.8. Training Time and Memory Costs

In this section, we compare the training time and memory cost of our proposed method TSNN with two other most competitive baseline methods (i.e., SMAN and EBGCN). In Table 6, we report these methods' training time and memory costs.

For the consumption of the training time, we can observe that SMAN takes less training time than other methods on all datasets. EBGCN shows the highest consumption of training time on all datasets. For our proposed method TSNN, its training time cost is lower than EGGCN and slightly higher than SMAN.

Table 6
Analysis of the training time and memory costs of different models.

	Twitter15		Twitter16		Weibo	
Method	Time	Memory	Time	Memory	Time	Memory
SMAN	429s	1987 M	179s	1641 M	1347s	5545 M
EBGCN	861s	2271 M	226s	2643 M	15480s	23895 M
TSNN	721s	2149 M	188s	1879 M	2568s	4537 M

For the cost of model memory, the baseline SMAN exhibits the lowest memory cost on the Twitter15 and Twitter16 datasets. On the Weibo dataset, TSNN shows the lowest memory cost. EBGCN shows the highest memory cost on all datasets. The memory cost of our method TSNN is much lower than that of EBGCN and is competitive with SMAN. Based on the analysis, TSNN has moderate training time and memory cost and can be applied to rumor detection.

4.9. Visualization

In order to examine the quality of our model in detecting rumors, we use t-SNE tool [38] to visualize the learned source post representations of our method and two state-of-the-art baselines (i.e., GLAN and SMAN), in which nodes are colored based on the ground-truth labels. As shown in Fig. 5, we can observe that our TSNN learns more discriminable source post representations. On the twitter based datasets, i.e., Twitter15 and Twitter16, GLAN mixes nodes with different class labels and cannot distinguish the source post categories well. On the Weibo dataset, although GLAN can distinguish most nodes, there is still a considerable number of nodes mixed.

Compared to GLAN, the source post representations learned by SMAN are more discriminable on the Twitter15 dataset. While on the datasets Twitter16 and Weibo, the intra-class similarity is not high enough. TSNN can learn more compact node representation with high intra-class similarity on all datasets. This illustrates that incorporating the topic distribution of source posts learned by NTM for enhancing the source post representation and introducing the topic credibility as weakly supervised information can facilitate our TSNN better representations for the task of rumor detection.

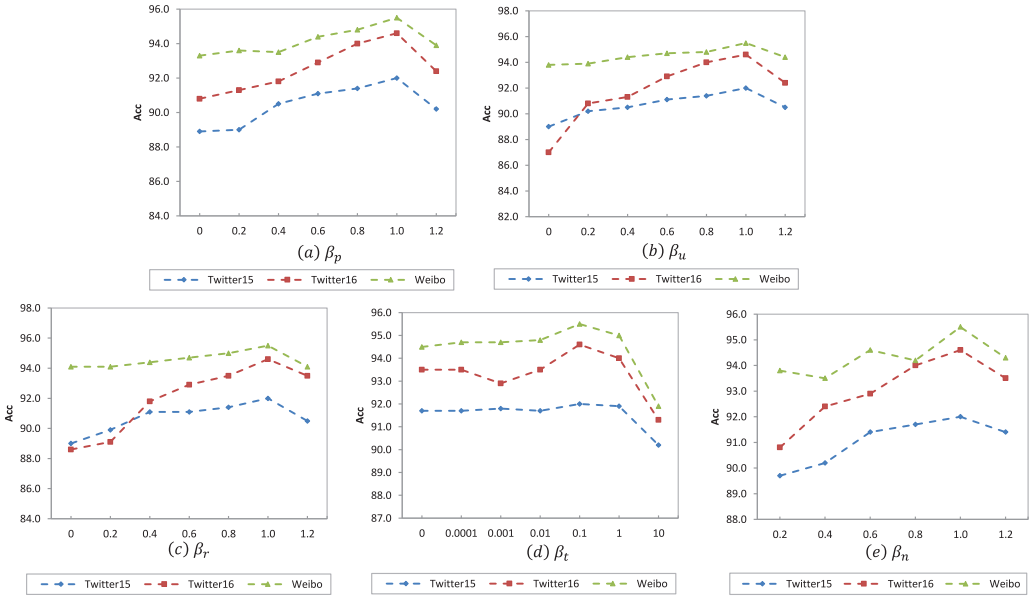

Fig. 4. Analysis of parameters.

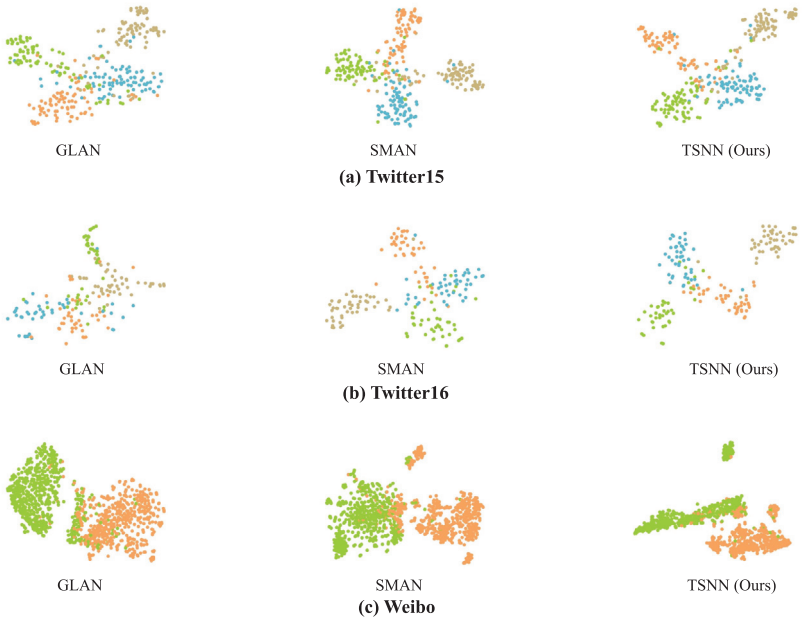

Fig. 5. Visualization of the learned source post representations of our method TSNN and two state-of-the-art baselines (i.e., GLAN and SMAN) on all three datasets. Nodes are colored based on the ground-truth labels.

5. Conclusion

In this paper, we introduce a new framework for rumor detection. Two kinds of topic signals, including a coarse-grained topic signal (i.e., topic credibility) and a fine-grained topic signal (i.e., latent topic representation), are explored to improve the performance of rumor detection. Specifically, the coarse-grained topic signal serves as the weakly supervised information to guide the learning process of source post representation, and the fine-grained topic signal is leveraged to better model the source post content by further exploring its underlying topic distribution. We evaluate the performance of our proposed model on three widely used datasets and compare them with current state-of-the-art baseline methods. Experimental results show that our model has a better performance than all baseline methods. In the future, we will explore how to boost the detection performance by incorporating the spatial and temporal information in rumor propagation.

CRediT authorship contribution statement

Zhuomin Chen: Conceptualization, Methodology, Software, Writing - original draft. **Li Wang:** Writing - original draft, Writing - review & editing, Supervision. **Xiaofei Zhu:** Writing - review & editing, Supervision. **Stefan Dietze:** Writing - review & editing, Supervision.

Data availability

The raw datasets can be downloaded from https://www.dropbox.com/s/46r50ctrfa0ur1o/rumdect.zip?dl=0. and https://www.dropbox.com/s/7ewzdrbelpmrnxu/rumdetect2017.zip?dl=0.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Key Research and Development Program of China [Grant No. 2021YFB3300503]; the Regional Innovation and Development Joint Fund of NSFC [Grant No. U22A20167]; the Natural Science Foundation of Chongqing, China [Grant No. CSTB2022NSCQ-MSX1672]; the Major Project of Science and Technology Research Program of Chongqing Education Commission of China [Grant No. KJZD-M202201102]; the Federal Ministry of Education and Research [Grant No. 01IS21086].

References

[1] V. Pérez-Rosas, B. Kleinberg, A. Lefevre, R. Mihalcea, Automatic detection of fake news, CoRR abs/1708.07104.
[2] S. Kwon, M. Cha, K. Jung, W. Chen, Y. Wang, Prominent features of rumor propagation in online social media, in: 2013 IEEE 13th International Conference on Data Mining, Dallas, 2013, pp. 1103–1108.
[3] F. Yang, Y. Liu, X. Yu, M. Yang, Automatic detection of rumor on sina weibo, in: Proceedings of the ACM SIGKDD workshop on mining data semantics, 2012, pp. 1–7.
[4] Z. Wang, Y. Guo, Rumor events detection enhanced by encoding sentimental information into time series division and word representations, Neurocomputing 397 (2020) 224–243.
[5] T. Ma, H. Zhou, Y. Tian, N. Al-Nabhan, A novel rumor detection algorithm based on entity recognition, sentence reconfiguration, and ordinary differential equation network, Neurocomputing 447 (2021) 224–234.
[6] J. Ma, W. Gao, K. Wong, Rumor detection on twitter with tree-structured recursive neural networks, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, 2018, pp. 1980–1989.
[7] Y. Wang, L. Wang, Y. Yang, T. Lian, Semseq4fd: Integrating global semantic relationship and local sequential order to enhance text representation for fake news detection, Expert Systems with Applications 166 (2021).
[8] Y. Wang, L. Wang, Y. Yang, Y. Zhang, Detecting fake news by enhanced text representation with multi-edu-structure awareness, Expert Systems with Applications 206 (2022).
[9] J. Ma, W. Gao, P. Mitra, S. Kwon, B.J. Jansen, K. Wong, M. Cha, Detecting rumors from microblogs with recurrent neural networks, in: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, 2016, pp. 3818–3824.
[10] J. Ma, W. Gao, K. Wong, Detect rumors in microblog posts using propagation structure via kernel learning, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017, pp. 708–717.
[11] Z. Jin, J. Cao, Y. Zhang, J. Luo, News verification by exploiting conflicting social viewpoints in microblogs, in: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, 2016, pp. 2972–2978.

[12] L. Hu, T. Yang, L. Zhang, W. Zhong, D. Tang, C. Shi, N. Duan, M. Zhou, Compare to the knowledge: Graph neural fake news detection with external knowledge, in: Proceedings of the 59th Annual Meeting of the Association for Computationa Linguistics and the 11th International Joint Conference on Natural Language Processing, (Volume 1: Long Papers), 2021, pp. 754–763.
[13] C. Yuan, Q. Ma, W. Zhou, J. Han, S. Hu, Jointly embedding the local and global relations of heterogeneous graph for rumor detection, in: 2019 IEEE International Conference on Data Mining, 2019, pp. 796–805.
[14] C. Yuan, Q. Ma, W. Zhou, J. Han, S. Hu, Early detection of fake news by utilizing the credibility of news, publishers, and users based on weakly supervised learning, in: Proceedings of the 28th International Conference on Computational Linguistics, 2020, pp. 5444–5454.
[15] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent dirichlet allocation, Journal of Machine Learning Research 3 (2003) 993–1022.
[16] Y. Miao, E. Grefenstette, P. Blunsom, Discovering discrete latent topics with neural variational inference, in: Proceedings of the 34th International Conference on Machine Learning, Vol. 70, 2017, pp. 2410–2419.
[17] Y. Wang, J. Li, H.P. Chan, I. King, M.R. Lyu, S. Shi, Topic-aware neural keyphrase generation for social media language, in: Proceedings of the 57th Conference of the Association for Computational Linguistics, 2019, pp. 2516–2526.
[18] L. Wang, Development and prospect of false information detection on social media, Journal of Taiyuan University of Technology 53 (3) (2022) 397–404.
[19] H. Ahmed, I. Traore, S. Saad, Detection of online fake news using n-gram analysis and machine learning techniques, in: International conference on intelligent, secure, and dependable systems in distributed and cloud environments, 2017, pp. 127–138.
[20] V. Agarwal, H.P. Sultana, S. Malhotra, A. Sarkar, Analysis of classifiers for fake news detection, Procedia Computer Science 165 (2019) 377–383.
[21] O. Ajao, D. Bhowmik, S. Zargari, Sentiment aware fake news detection on online social networks, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2019, pp. 2507–2511.
[22] K. Popat, Assessing the credibility of claims on the web, in: Proceedings of the 26th International Conference on World Wide Web Companion, 2017, pp. 735–739.
[23] J. Devlin, R. Zbib, Z. Huang, T. Lamar, R.M. Schwartz, J. Makhoul, Fast and robust neural network joint models for statistical machine translation, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, 2014, pp. 1370–1380.
[24] K. Cho, B. van Merrienboer, D. Bahdanau, Y. Bengio, On the properties of neural machine translation: Encoder-decoder approaches, in: Proceedings of SSST@EMNLP 2014, 2014, pp. 103–111.
[25] V. Vaibhav, R.M. Annasamy, E.H. Hovy, Do sentence interactions matter? leveraging sentence level representations for fake news classification, in: Proceedings of the Thirteenth Workshop on Graph-Based Methods for Natural Language Processing, 2019, pp. 134–139.
[26] F. Yu, Q. Liu, S. Wu, L. Wang, T. Tan, A convolutional approach for misinformation identification, in: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, 2017, pp. 3901–3907.
[27] T. Chen, X. Li, H. Yin, J. Zhang, Call attention to rumors: Deep attention based recurrent neural networks for early rumor detection, in: Pacific-Asia conference on knowledge discovery and data mining, Vol. 11154, 2018, pp. 40–52.
[28] Y. Liu, Y.B. Wu, 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, 2018, pp. 354–361.
[29] L. Wei, D. Hu, W. Zhou, Z. Yue, S. Hu, Towards propagation uncertainty: Edge-enhanced bayesian graph convolutional networks for rumor detection, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, 2021, pp. 3845–3854.
[30] Y. Kim, Convolutional neural networks for sentence classification, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, 2014, pp. 1746–1751.
[31] D.P. Kingma, M. Welling, Auto-encoding variational bayes, in: 2nd International Conference on Learning Representations, 2014.
[32] C. Castillo, M. Mendoza, B. Poblete, Information credibility on twitter, in: Proceedings of the 20th International Conference on World Wide Web, 2011, pp. 675–684.
[33] J. Ma, W. Gao, Z. Wei, Y. Lu, K. Wong, Detect rumors using time series of social context information on microblogging websites, in: Proceedings of the 24th ACM International Conference on Information and Knowledge Management, 2015, pp. 1751–1754.
[34] Z. Zhao, P. Resnick, Q. Mei, Enquiring minds: Early detection of rumors in social media from enquiry posts, in: Proceedings of the 24th International Conference on World Wide Web, 2015, pp. 1395–1405.
[35] S. Kwon, M. Cha, K. Jung, Rumor detection over varying time windows, PloS one 12 (1).
[36] S.J. Reddi, S. Kale, S. Kumar, On the convergence of adam and beyond, in: 6th International Conference on Learning Representations, 2018.
[37] X. Zhu, Z. Peng, J. Guo, D. Stefan, Generating effective label description for label-aware sentiment classification, Expert Systems with Applications 119194 (2022).
[38] V.D.M. Laurens, G. Hinton, Visualizing data using t-sne, Journal of Machine Learning Research 9 (86) (2008) 2579–2605.

Zhuomin Chen is a Master Candidate at the College of Data Science, Taiyuan University of Technology. She received her B.S. degree in Applied Statistics from Qingdao University in 2020. Her main research interest focuses on machine learning and text mining.

Prof. Dr. Li Wang is a professor, Ph.D advisor in the College of Data Science, Taiyuan University of Technology. She is ACM, CCF senior member of China computer society, standing committee of CCF big data expert committee, expert committee of CCF artificial intelligence and pattern recognition, member of CCF collaborative computing Committee and service intelligence committee of China AI society. Her main research fields are big data computing, machine learning, artificial intelligence, etc. She has undertaken more than 30 projects and published more than 100 academic papers. She was the program committee of many conferences and hosted and served for many conferences.

Prof. Dr. Xiaofei Zhu is a full professor at College of Computer Science and Engineering, Chongqing University of Technology. He received his PhD degree at the Institute of Computing Technology, Chinese Academy of Science (ICT-CAS) in 2012. Then he spent four years as a Postdoctoral Research Fellow at the L3S Research Center, Leibniz University Hannover. His research interests include web search, data mining and machine learning, and he has published more than 30 papers in international conferences and journals, including the top conferences like SIGIR, WWW, CIKM, TKDE, etc. He has won the Best Paper Awards of CIKM (2011). He serves as area chair, program committees and editorial board of numerous international conferences and journals, including SIGIR, AAAI, CIKM, etc.

Prof. Dr. Stefan Dietze is a professor at the University of Düsseldorf and the scientific director of Knowledge Technologies for the Social Sciences (WTS), GESIS. He received his Ph.D. from the Institute for Computer Science of the University of Potsdam (Ph.D./Dr. rer. nat. in Applied Computer Science), Germany, in 2004. His professional career led him to the Fraunhofer Institute for Software and Systems Engineering (ISST) in Berlin. Then, he spent five years as a Postdoctoral Research Fellow at the Knowledge Media Institute (KMi) of The Open University in Milton Keynes, UK. Before joining GESIS, he led a research group at the L3S research center at Leibniz University Hannover. His research interests include semantic web technologies, web intelligence, semantic search, and information retrieval. He has published numerous papers in prestigious journals and international conferences, including SIGIR, WWW, CIKM, ESWC, CHIIR and so on. He has been general/-poster/track chair, program committee and editorial board member of numerous international conferences and journals.