

## 传播树结构结点及路径双注意力谣言检测模型

韩雪明, 贾彩燕, 李轩涯, 张鹏飞

引用本文

韩雪明, 贾彩燕, 李轩涯, 张鹏飞. 传播树结构结点及路径双注意力谣言检测模型[J]. 计算机科学, 2023, 50(4): 22-31.

HAN Xueming, JIA Caiyan, LI Xuanya, ZHANG Pengfei. [Dual-attention Network Model on Propagation Tree Structures for Rumor Detection](#) [J]. Computer Science, 2023, 50(4): 22-31.

---

## 相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[基于合成图像和Xception改进模型的安卓恶意家族分类方法](#)

Android Malware Family Classification Method Based on Synthetic Image and Xception Improved Model

计算机科学, 2023, 50(4): 351-358. <https://doi.org/10.11896/jsjcx.220300200>

[基于抽象语法树裁剪的智能合约漏洞检测研究](#)

Smart Contract Vulnerability Detection Based on Abstract Syntax Tree Pruning

计算机科学, 2023, 50(4): 317-322. <https://doi.org/10.11896/jsjcx.220300063>

[基于交互注意力和图卷积网络的方面级情感分析](#)

Aspect-level Sentiment Classification Based on Interactive Attention and Graph Convolutional Network

计算机科学, 2023, 50(4): 196-203. <https://doi.org/10.11896/jsjcx.220100105>

[基于用户长短期偏好的序列推荐模型](#)

Sequential Recommendation Model Based on User's Long and Short Term Preference

计算机科学, 2023, 50(4): 47-55. <https://doi.org/10.11896/jsjcx.220100264>

[基于多邻接图与多头注意力机制的短期交通流量预测](#)

Short-time Traffic Flow Forecasting Based on Multi-adjacent Graph and Multi-head Attention Mechanism

计算机科学, 2023, 50(4): 40-46. <https://doi.org/10.11896/jsjcx.220200079>

# 传播树结构结点及路径双注意力谣言检测模型

韩雪明<sup>1,2</sup> 贾彩燕<sup>1,2</sup> 李轩涯<sup>3</sup> 张鹏飞<sup>1,2</sup>

1 北京交通大学计算机与信息技术学院 北京 100044

2 北京交通大学交通数据分析与挖掘北京市重点实验室 北京 100044

3 百度在线网络技术(北京)有限公司 北京 100085

(xueminghan@bjtu.edu.cn)

**摘要** 随着社交媒体平台的快速发展和移动设备的普及,人与人之间的交互变得更加便捷。但同时,谣言在社交媒体上也更加肆虐,给公众和社会安全带来巨大的隐患。在现实世界中,用户在发表自己的评论之前,往往会首先观测其他已经发表的帖子,尤其是即将评论的帖子上下文。现有的一些谣言检测方法虽然使用了谣言传播过程中的传播结构,基于群体智能原则提取用户间的相互质疑或事实线索,极大地提高了谣言检测的效果,但是对传播结构的深层非直接隐式关系及关键帖子和关键路径重要性的学习能力不足。据此,文中提出了一种基于传播树的结点及路径双注意力谣言检测模型 DAN-Tree(Dual-attention Network Model on Propagation Tree Structures)。该模型使用 Transformer 结构学习传播路径中帖子间的隐式语义关系,并利用注意力机制学习路径中结点的重要性;其次,利用注意力机制对路径表示进行加权聚合得到整个传播树的表示向量;最后,基于传播树表示向量进行谣言分类。此外,我们使用结构嵌入方法学习帖子在传播树上的空间位置信息,进一步对谣言传播结构上的深层结构和语义信息进行融合。在 4 个经典数据集上的实验结果表明:DAN-Tree 模型在其中的 3 个数据集上都超过了目前已有文献的最优结果。在 Twitter15 和 Twitter16 数据集上正确率指标分别提升了 1.81% 和 2.39%,在 PHEME 数据集上  $F_1$  指标提升了 7.51%。

**关键词:** 谣言检测;传播结构;注意力机制;社交媒体;深度学习

**中图法分类号** TP181

## Dual-attention Network Model on Propagation Tree Structures for Rumor Detection

HAN Xueming<sup>1,2</sup>, JIA Caiyan<sup>1,2</sup>, LI Xuanyan<sup>3</sup> and ZHANG Pengfei<sup>1,2</sup>

1 School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China

2 Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University, Beijing 100044, China

3 Baidu Online Network Technology(Beijing)Co., Ltd., Beijing 100085, China

**Abstract** With the rapid development of social media and the popularity of mobile devices, the interaction between users has become more convenient. But at the same time, rumors on social media are more and more rampant, which brings hidden dangers to the public and social safety. In the real world, users often express their own opinions after observing other microblogs that have been posted, especially the context of the microblog to be replied. Although some existing rumor detection methods learn the propagation patterns on propagation trees of rumors to extract clues of user interrogation or factual evidences based on the principle of crowd wisdom, which greatly improves the performance of rumor detection, they only focus on those microblogs that have direct response relationships, and Lack of ability to fully mine the indirect and implicit relationships among microblogs in the process of rumor propagation. Therefore, in this paper, a node and path dual-attention network on propagation tree structures (DAN-Tree) for debunking rumors is proposed. First, the model uses the Transformer structure to fully learn the implicit semantic relationship between posts in the propagation path, and then uses the attention mechanism to perform weighted fusion to obtain the feature vector of the propagation path. Secondly, the path representation is weighted and aggregated by using the attention mechanism to obtain the representation vector of the whole propagation tree. In addition, the structure embedding method is used to learn the spatial location information of the post on the propagation tree, which realizes the effective fusion of the deep structure and semantic information in the rumor propagation structure. The effect of the DAN-Tree model is verified on four clas-

到稿日期:2022-02-08 返修日期:2022-05-12

基金项目:国家重点研发计划(2017YFC1703506);国家自然科学基金(61876016);中央高校基本科研业务费专项资金(2019JBZ110);百度松果计划开放研究基金

This work was supported by the National Key R & D Program of China (2017YFC1703506), National Natural Science Foundation of China (61876016), Fundamental Research Funds for the Central Universities(2019JBZ110) and Open Research Fund of Baidu Pinecone Program.

通信作者:贾彩燕(cyjia@bjtu.edu.cn)

sic datasets. Experimental results show that the DAN-Tree model surpasses the best results of the existing literature on the three datasets; the accuracy of the Twitter15 and Twitter16 datasets increases by 1.81% and 2.39%, respectively, and the  $F_1$  score of the PHEME dataset increases by 7.51%, which proves the effectiveness of DAN-Tree model.

**Keywords** Rumor detection, Propagation structure, Attention mechanism, Social media, Deep learning

## 1 引言

谣言(Rumor),也被称为流言,是人们广泛关注但其真实性未经证实的阐述<sup>[1]</sup>。这种阐述往往被用来有意误导大众并迅速扩大舆论影响,阐述为假的谣言也会因此造成公众恐慌和不安情绪的蔓延。随着科技的发展和社交媒体的广泛应用,人们在互联网上的交互程度达到了一种前所未有的高度。每个人都可以非常便捷地在社交平台上表达各自的意见和观点。然而这种社交的便捷性也为大量谣言的广泛传播提供了便利。同时,在快节奏而且普遍缺乏专业知识判断的当下,人们非常容易相信并传播一条信息,即使这条信息是发表者有意误导大众的,这也加剧了谣言的迅速传播并带来潜在的危害。因此,构建一个有效的自动谣言检测方法非常必要。

根据 Zubiaga 等<sup>[2]</sup>提出的定义,一个完整的谣言分类系统由4个子任务组成:谣言检测、谣言追踪、立场分类和谣言验证。其中,谣言检测和谣言验证两个子任务在目标上很接近。谣言检测是一个二分类任务,旨在检测出一个帖子是谣言(Rumor)还是非谣言(Non-rumor)。而谣言验证是一个多分类的任务,它的目标是判断一个已经被检测为谣言的帖子的真实性标签,包括:真实(True)、错误(False)和未经验证(Unverified)。不同于以往仅仅检测帖子事件是否为谣言的工作, Ma 等<sup>[3]</sup>将谣言检测和谣言验证这两个子任务联合起来作为一个新的多分类任务(Non, True, False, Unverified),并将这个联合任务用“谣言检测”来描述,本文沿用了这种定义。

Zubiaga 等<sup>[2]</sup>提出帖子可以通过用户之间分享观点、猜测和证据来进行“自我纠错”。因此,在谣言传播过程中利用传播结构上用户之间的评论可以提供对谣言检测任务有益的信息。 Wu 等<sup>[4]</sup>也证明了传播结构在谣言检测领域是有效的。 Ma 等<sup>[5]</sup>提出的 RvNN 模型使用递归神经网络对谣言传播树结构进行自顶向下和自底向上的建模,将语义信息和结构信息有机地结合起来。 Bian 等<sup>[6]</sup>提出的 Bi-GCN 模型使用 GCN(Graph Convolutional Networks)从自顶向下和自底向上两个方向对谣言的传播和扩散特征进行建模。这些基于空间树结构传播路径建模的模型可以对帖子事件在真实传播情况下的传播结构进行建模,进而捕捉到帖子事件的结构信息,极大地提高了谣言检测的准确性。

但是,对于一个在现实世界中不断扩散的谣言传播事件而言,从时间尺度上来说,用户往往会首先观测大部分已经发表的帖子之后才发表自己的回复,尤其会重点关注将要回复的帖子的上下文。这种现象给传播路径中所有的帖子之间建立了一个隐式的间接关系。而上述利用传播结构的研究大部分过多地关注了帖子之间显式的近距离直接关系,而忽略了这种远距离隐式关系,导致模型学习到的帖子表示可能过度简化了用户间的交互性。此外,这些方法对中间路径上结点

和不同路径的重要性缺少学习能力。

近年来, Vaswani 等<sup>[7]</sup>提出的 Transformer 结构使用注意力机制对单词的远距离交互进行建模,为顺序序列中的任意单词对建立联系。这种结构在机器翻译<sup>[8]</sup>、语言模型<sup>[9]</sup>和情感分析<sup>[10]</sup>等诸多自然语言处理任务中都取得了非常好的效果。在谣言检测领域, Khoo 等<sup>[11]</sup>提出的 PLAN 模型将谣言事件建模为时间序列,并将 Transformer 结构应用于时间序列上的语义提取,使得谣言传播路径中的帖子可以关注到与其他远距离帖子间的联系,增强了用户间的交互性。但是,该模型在结构信息的利用方面,仅在时间序列建模的基础上简单地使用了一个用于描述帖子间回复类型的变量,不能获取传播树结构上的全局关系,使得模型不能有效利用谣言传播结构中具有指向性线索的深层结构信息。

本文针对上述存在的谣言检测问题提出了一种基于传播树结构的结点和路径双注意力网络模型 DAN-Tree。该模型直接在传播树结构中利用 Transformer 架构学习帖子的语义关系,使模型不但能关注到传播路径中帖子之间的近距离显式关系,还能关注到帖子之间的非直接隐式关系,更好地利用了用户之间互相讨论、支持和质疑等行为产生的互动语义信息,从而提取出这些互动中包含的对帖子真实性的指向性线索。同时本文使用两次注意力机制,学习帖子结点的重要性和路径的重要性。此外,本文使用了结构嵌入<sup>[8]</sup>的方法学习帖子结点在传播树上的空间位置信息,弥补 Transformer 结构对空间位置信息不敏感的缺陷,使模型能更好地学习谣言传播树上的深层结构和语义信息。我们在具有传播结构的4个谣言数据集(Twitter15, Twitter16, Weibo, PHEME)上验证了所提模型的有效性。

## 2 相关工作

早期的检测方法主要是从帖子事件中提取各种各样的判别性特征,利用传统机器学习方法检测帖子的真实性。例如,决策树方法<sup>[12-13]</sup>、随机森林方法<sup>[14]</sup>和支持向量机(Support Vector Machine, SVM)方法<sup>[15-16]</sup>等。这些人工设计的特征或半自动化特征分析提取出的特征非常耗时耗力,而且难以提取到谣言事件的高层特征信息,在谣言检测任务上的效果不尽人意。

近年来,受深度学习方法在各领域(特别是自然语言处理领域)被广泛应用并且取得显著效果的影响,出现了很多在谣言检测领域应用各种各样神经网络的研究。文献<sup>[17]</sup>将微博事件按照时间段进行分割,并输入带有注意力机制的双向 GRU(Gate Recurrent Unit)网络,获取时间段内微博序列的隐层表示。 Ma 等<sup>[18]</sup>使用循环神经网络(Recurrent Neural Networks, RNN)学习帖子事件在时间上的隐层特征。 Yu 等<sup>[19]</sup>使用卷积神经网络(Convolutional Neural Networks, CNN)学习输入序列中的关键特征。 Liu 等<sup>[20]</sup>对时间序列上

的用户特征联合使用 RNN 和 CNN 进行建模。Guo 等<sup>[21]</sup>在堆叠 RNN 结构的基础上使用注意力机制对帖子事件的时间传播特征进行建模。Yuan 等<sup>[22]</sup>将所有源帖子和转发帖子根据转发关系连接起来,并添加了发表帖子的用户结点,将数据集构成一张异构图,用来学习文本的全局语义关系和局部语义关系。

与提取时间序列上文本特征的方法相比,传播树结构中的传播路径包含了用户之间对帖子事件的互相争论信息,有助于提取到帖子事件的真实性线索。Ma 等<sup>[5]</sup>使用递归神经网络对谣言传播树结构进行自顶向下或自底向上的建模,融合传播过程的结构信息和文本的语义信息。Kumar 等<sup>[23]</sup>使用 Tree-LSTM(Long Short-Term Memory)结构对传播树结构进行显式建模。这两种经典方法利用了谣言传播过程中的树状结构进行建模,充分证明了传播结构信息对谣言检测具有巨大的帮助。Ma 等<sup>[24]</sup>在对谣言传播树结构进行自顶向下的递归建模之后,对所有帖子结点使用全局注意力机制,关注不同结点对谣言表示的不同重要度。Ma 等<sup>[25]</sup>从谣言传播树结构的根结点到叶子结点,对同一层的所有结点及其所属根结点使用 Transformer 结构,以聚合各个子树之内和子树之间的立场态度。Bian 等<sup>[6]</sup>使用双向 GCN 模型学习谣言传播结构的扩散和传播结构,得到了一个更好的谣言传播表示。然而这些方法大多关注于对近距离直接回复关系的帖子之间进行建模,忽略了时间尺度上各帖子间较长距离的隐式关系。

Khoo 等<sup>[11]</sup>在时间序列上使用 Transformer 结构,构建输入序列中任意帖子对的隐式关系,充分利用了用户间的交互

关系,但是过度简化了对传播结构信息的利用。因此,本文综合考虑了帖子间的隐式关系和谣言在传播过程中的树状结构,使用 Transformer 结构提取高层语义信息,充分利用传播路径中的用户交互信息,利用注意力机制学习不同帖子结点和不同传播路径的重要性,并引入结构嵌入方法学习结点的空间位置信息,将语义信息和结构信息进行融合得到更有效的谣言表示,以实现更好的谣言检测效果。

### 3 模型方法

为了在谣言事件的树状传播结构中学习到用户评论间的隐式关系,充分利用用户间的直接交互和间接交互,本文在谣言检测任务中提出了一种基于传播树结构的结点及路径双注意力网络模型 DAN-Tree。该模型主要包括 3 个部分:帖子级注意力模块、路径级注意力模块及谣言分类模块,模型的整体结构如图 1 所示。其中,帖子级注意力模块使用 Transformer 结构在传播树结构上学习帖子间的隐式语义关系,并利用注意力机制学习不同帖子表示对传播路径的不同重要度,同时使用结构嵌入的方法增强学习传播树结构上的位置关系信息;路径级注意力模块考虑到传播路径表示对谣言传播树具有不同的重要度,使用注意力机制来衡量不同传播路径表示的不同重要度并进行加权聚合得到整棵传播树结构的表示;谣言分类模块学习一个分类器对帖子事件进行标签预测。下文将从问题描述、Transformer 结构中的编码块、结构嵌入、路径过采样、帖子级注意力、路径级注意力、分类预测几个方面详细介绍本文提出的模型 DAN-Tree。

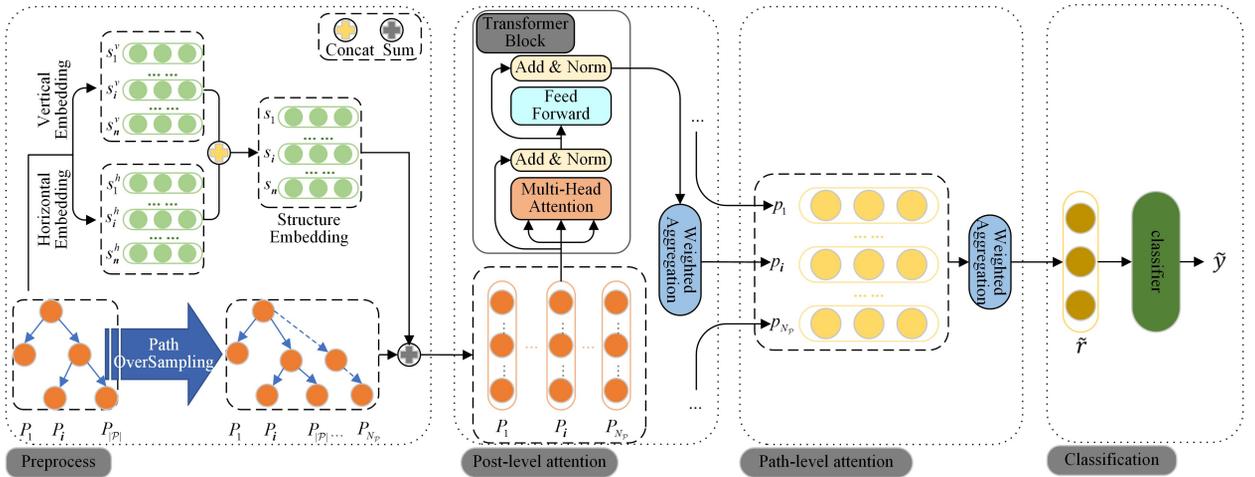


图 1 基于传播树结构的结点路径双注意力网络模型 DAN-Tree

Fig. 1 Node-path dual-attention network model based on propagation tree structure(DAN-Tree)

#### 3.1 问题描述

通常给定一个谣言事件  $E = \{T_1, T_2, \dots, T_{|E|}\}$  和对应的类别标签  $Y$ 。其中,  $T_1$  代表谣言事件  $E$  的源帖子,  $T_2, T_3, \dots, T_{|E|}$  是  $T_1$  根据时间顺序排列而成的所有相关回复帖子。类别标签  $Y$  有以下 4 种可能:非谣言(Non-Rumor, NR)、验证为假的谣言(False Rumor, FR)、验证为真的谣言(True Rumor, TR)和未经验证的谣言(Unverified Rumor, UR)。

对于一个谣言事件  $E$ ,根据事件在真实情况下的传播树结构信息,我们可以将  $E$  重新构造为一个由多条传播路径组成的传播树:  $\mathcal{P} = \{P_1, P_2, \dots, P_{|\mathcal{P}|}\}$ 。其中,  $|\mathcal{P}|$  是事件所拥有

的传播路径数目,  $P_i$  是事件  $E$  的第  $i$  条传播路径,且  $P_i = \{T_{i1}, T_{i2}, \dots, T_{i|P_i|}\}$ ;  $T_{i1}$  是路径  $P_i$  的根结点,也是整个事件的根结点;  $T_{i|P_i|}$  是路径  $P_i$  的叶子结点,  $P_i$  是由从  $T_{i1}$  到  $T_{i|P_i|}$  上所有具有回复关系的帖子构成的一条传播路径。

谣言检测任务是学习一个合适的分类器  $f: \{E \rightarrow Y\}$ ,即对于一个谣言事件  $E$ ,通过分类器  $f$  将其映射为该事件对应的真实标签  $Y$ 。

#### 3.2 Transformer 结构中的编码块 (Block)

Transformer<sup>[7]</sup>是一种用于序列学习任务的前向注意力编解码网络模型,它无需传统 RNN 的循环结构和 CNN 的

卷积结构,仅使用注意力机制便能对输入输出的全局依赖关系进行建模,解决了RNN无法并行运算的问题,缓解了RNN顺序计算产生的长距离依赖信息丢失现象,受到了极大的关注并被成功应用于自然语言处理<sup>[26]</sup>、图像文字识别<sup>[27-28]</sup>等诸多序列学习任务。

Transformer结构的编码模块由多个编码块(Block)堆叠而成,每个编码块(Block)由多头注意力层(Multi-Head Attention)和前馈网络层(Feed-Forward Network)两个子层构成。Transformer结构的核心机制自注意力(Self-attention)由式(1)计算得到。

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \quad (1)$$

其中,  $\mathbf{Q} \in \mathbb{R}^{l_q \times d}$ ,  $\mathbf{K} \in \mathbb{R}^{l_k \times d}$ ,  $\mathbf{V} \in \mathbb{R}^{l_v \times d}$  分别对应着序列输入  $\mathbf{X} \in \mathbb{R}^{l \times d}$  的 *query*, *key* 及 *value* 矩阵且通常可令  $\mathbf{Q} = \mathbf{K} = \mathbf{V} = \mathbf{X}$ , softmax 代表 softmax 层。通过 attention 操作可以得到序列输入中每个位置的相似度得分矩阵。

多头注意力机制将模型分为  $h$  个头,形成了多个表示子空间。它将序列输入在  $h$  个子空间的表示信息拼接起来,允许模型关注来自不同方面的表示信息,其计算式如式(2)所示:

$$\text{MHA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \dots, \text{head}_i) \mathbf{W}^O \quad (2)$$

$$\text{head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V)$$

其中,  $\mathbf{W}_i^Q \in \mathbb{R}^{d \times d_q}$ ,  $\mathbf{W}_i^K \in \mathbb{R}^{d \times d_k}$ ,  $\mathbf{W}_i^V \in \mathbb{R}^{d \times d_v}$  和  $\mathbf{W}^O \in \mathbb{R}^{hd_v \times d}$  表示可训练的权重矩阵,且  $d_q = d_k = d_v = d/h$ ,  $\text{head}_i$  表示序列

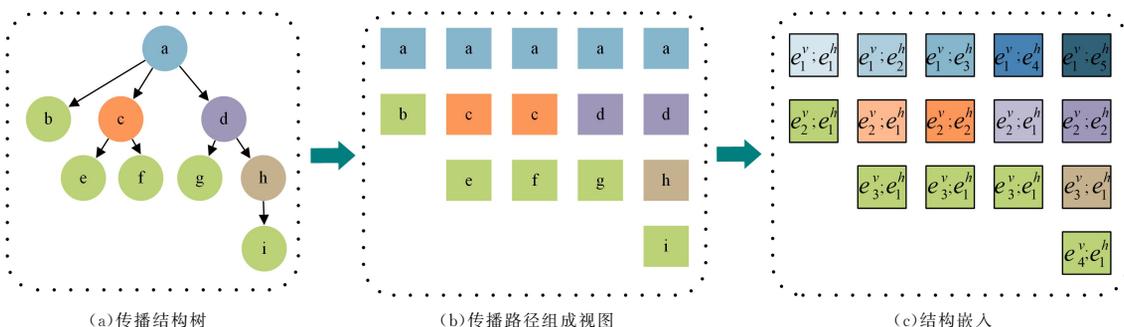


图2 结构嵌入过程

Fig. 2 Structure embedding process

图2(a)给出了一棵谣言传播树结构,每个结点都表示一个现实中的帖子结点。图2(b)是图2(a)对应的传播路径组成的路径视图。根据图2(b)所示的路径视图,我们可以从水平和垂直两个视角编码结点在树结构中的位置信息,即可以通过式(5)对路径中的每个帖子结点计算一个相应的结构嵌入表示,如图2(c)所示。

$$\mathbf{s}_{ij} = [\mathbf{e}_x^v; \mathbf{e}_y^h], x = |V_j^i| \text{ and } y = |H_j^i| \quad (5)$$

其中,  $\mathbf{s}_{ij} \in \mathbb{R}^d$  代表传播树的路径  $P_i$  中帖子结点  $T_{ij}$  的结构嵌入,  $\mathbf{e}_x^v, \mathbf{e}_y^h \in \mathbb{R}^{d/2}$  分别代表可训练的垂直位置嵌入矩阵  $\mathbf{E}^v \in \mathbb{R}^{h_p \times d/2}$  和水平位置嵌入矩阵  $\mathbf{E}^h \in \mathbb{R}^{|\mathcal{P}| \times d/2}$  中第  $x$  和第  $y$  个行向量,  $[\cdot; \cdot]$  表示拼接操作,  $h_p$  代表  $|\mathcal{P}|$  条传播路径中深度的最大值,  $x = |V_j^i|$  代表  $T_{ij}$  在所属传播路径中的垂直位置信息(自顶向下编码位置),故  $V_j^i = \{T_k | 1 \leq k \leq j\}$  代表信息从根结点  $T_{i1}$  流向结点  $T_{ij}$  的过程中途径的结点集合。  $y = |H_j^i|$  代表  $T_{ij}$  在当前传播路径时共出现过的次数(从左向

输入在第  $i$  个子空间的表示信息,  $\text{Concat}$  表示拼接操作。

多头注意力层的输出  $\mathbf{O}$  由式(3)计算得到。

$$\mathbf{O} = \text{LayerNorm}(\mathbf{Q} + \text{MHA}(\mathbf{Q}, \mathbf{K}, \mathbf{V})) \quad (3)$$

其中,  $\text{LayerNorm}$  代表模型中的归一化层。

Transformer结构的编码块的最终输出如式(4)所示:

$$\text{Trans}(\mathbf{X}) = \text{LayerNorm}(\mathbf{O} + \text{FFN}(\mathbf{O})) \quad (4)$$

其中,  $\mathbf{X}$  表示模型的序列任务输入,  $\text{FFN}$  表示模型中的前馈网络层。

将多个这样的编码块进行堆叠可以有效学习整个序列的表示向量并关注那些在重要作用位置上的向量。该基础结构常被用于序列预测任务,构成了本文谣言传播路径上序列文本语义提取的基础模块。

### 3.3 结构嵌入

Transformer结构已经在许多领域的序列任务中取得了非常显著的效果,但是它天然地难以处理树状结构数据。针对这种问题,Nguyen等<sup>[8]</sup>提出了一种层次嵌入的方法,该方法在模型中引入树状结构在各条路径内部的层次信息和路径之间的兄弟关系信息,可以对路径中的结点位置及路径深度信息进行位置编码。同时,在一棵谣言传播树中,存在着信息从根结点到叶子结点的流向。因此,为了处理谣言事件的树状传播结构数据,我们将Nguyen等<sup>[8]</sup>提出的层次位置嵌入应用在谣言检测任务中,使其可以在模型中学习谣言传播结构在空间上的水平位置信息和垂直位置信息,本文称之为结构嵌入,如图2所示。

右编码),即在树结构中的水平位置信息,其中  $H_j^i = \{T_{kj} | 1 \leq k \leq i \text{ and } T_{kj} = T_{ij}\}$  代表  $T_{ij}$  在传播路径  $P_i$  及其之前的所有路径中出现的集合。

综上,对于树中的任意一个结点,我们可以由一个水平位置向量和一个垂直位置向量编码其在树中的结构信息以刻画整个树结构信息。

### 3.4 路径过采样

不同传播树结构拥有不同的传播路径数量,数量较少的传播路径可能会难以得到足够有用的用户反馈。为了解决这一问题,我们提出路径过采样的方法,通过随机重采样原始的变长传播路径序列  $\mathcal{P}$ ,得到一个固定长度为  $N$  的定长传播路径序列  $\mathcal{R} = \{R_1, R_2, \dots, R_{|\mathcal{P}|}, \dots, R_{N_p}\}$ 。同时,深度较深的传播路径通常会有更多的用户参与到对源结点的讨论中,由此得到一个对源结点更可信的反馈。因此我们在随机重采样的过程中加大对较深传播路径的采样概率。对于长度  $|\mathcal{P}|$  大于

$N_\rho$  的变长传播路径序列, 我们仅使用前  $N_\rho$  条传播路径。路径过采样的方法细节如算法 1 所示。

### 算法 1 路径过采样算法

输入: 原始的变长传播路径序列  $\mathcal{P}$

输出: 过采样之后的定长传播路径序列

1. 初始化一个空的路径下标集合  $I = \{\}$
2. FOR  $i = 1 \rightarrow N_\rho$
3. IF  $i \leq |\mathcal{P}|$  THEN
4.  $R_i \leftarrow P_i$
5. FOR  $j = 1 \rightarrow (2|P_i| - 3)$
6.  $I \leftarrow I \cup \{i\}$
7. END FOR
8. ELSE
9.  $j \leftarrow \text{Random}(I)$
10.  $R_i \leftarrow P_j$
11. END IF
12. END FOR

为方便描述, 定长传播路径序列  $\mathcal{R}$  在下文中仍用  $\mathcal{P}$  表示, 此时  $\mathcal{P} = \{P_1, P_2, \dots, P_{N_\rho}\}$ , 路径数量为固定的  $N_\rho$ 。

根据 3.3 节所述, 每个帖子结点  $T_{ij}$  都有一个相应的结构嵌入表示  $s_{ij}$ 。在路径过采样的过程中, 对于一个从原始路径  $P_i$  采样得到的路径  $P_k$ , 令其上面的帖子结点  $T_{kj}$  对应的结构嵌入为  $s_{kj} = s_{ij}$ 。

### 3.5 帖子级注意力

帖子  $T_{ij}$  由一系列单词组成, 可以将其表示为  $T_{ij} = \{W_{ij1}, W_{ij2}, \dots, W_{ij|T_{ij}|}\}$ 。其中,  $W_{ijk}$  代表  $T_{ij}$  中的第  $k$  个单词, 其对应的单词嵌入为  $w_{ijk} \in \mathbb{R}^d$ ,  $d$  是单词嵌入的维度。帖子  $T_{ij}$  对应的句子表示  $t_{ij}$  可由式(6)计算得到。

$$t_{ij} = \text{MaxPooling}(\{w_{ij1}, w_{ij2}, \dots, w_{ij|T_{ij}|}\}) \quad (6)$$

从传播路径的时间尺度上来说, 某个回复帖子  $T_{ij}$  不只受到它所直接回复的父结点的影响, 还很有可能受到所有较早发表的帖子 ( $T_k, 1 \leq k < j$ ), 尤其是根结点的影响。因此, 对于一条传播路径  $P_i$ , 为了学习到它所包含的各个帖子之间的隐式关系, 同时也能够对模型进行并行化训练, 我们通过式(8)对  $P_i$  上的帖子表示序列应用  $N_i$  层 Transformer 结构中的编码块。在此之前, 为了考虑到传播树结构在空间上的信息, 我们在原有的帖子表示  $t_{ij}$  上加入了式(5)所示的结构嵌入  $s_{ij}$ :

$$t_{ij} = t_{ij} + s_{ij} \quad (7)$$

$$\{t_{i1}, t_{i2}, \dots, t_{i|P_i|}\} = \text{Trans}(\{t_{i1}, t_{i2}, \dots, t_{i|P_i|}\}) \quad (8)$$

由于不同的帖子可能对所属传播路径的表示具有不同的重要度, 为了衡量这种重要度并加以利用, 我们使用式(9)所示的注意力方法得到帖子级的上下文向量  $u_i^p$ 。

$$u_i^p = a_i^T \cdot \text{LeakyReLU}(W_i t_{ij}) \quad (9)$$

其中,  $\text{LeakyReLU}$  作为激活函数,  $a_i$  代表权重向量,  $W_i$  表示权重矩阵。由式(10)得到归一化的重要度权重  $\beta_{ij}^p$ , 之后由式(11)对帖子表示进行加权求和得到传播路径  $P_i$  的表示  $p_i$ 。

$$\beta_{ij}^p = \frac{\exp(u_{ij}^p)}{\sum_{k=1}^{|P_i|} \exp(u_k^p)} \quad (10)$$

$$p_i = \sum_{j=1}^{|P_i|} \beta_{ij}^p t_{ij} \quad (11)$$

### 3.6 路径级注意力

显然, 在一棵谣言传播树结构中, 每个帖子所表达的内容并不一样, 因此由多个帖子组成的传播路径之间对整棵传播树的重要度也是不同的。不同于 Ma 等<sup>[5]</sup> 直接使用 Max-Pooling 方法或 Bian 等<sup>[6]</sup> 使用 MeanPooling 方法得到最终谣言表示的做法, 我们再次使用注意力机制, 引入路径级的上下文向量  $u_i^p$  来衡量路径表示  $p_i$  对传播树结构表示的不同重要度。在式(12)中我们仍然使用  $\text{LeakyReLU}$  函数作为激活函数,  $a_p$  代表权重向量,  $W_p$  表示权重矩阵。通过式(13)得到归一化的重要度权重  $\beta_i^p$ , 最后通过式(14)对路径表示进行加权求和得到传播树的表示  $\tilde{r}$ 。

$$u_i^p = a_p^T \cdot \text{LeakyReLU}(W_p p_i) \quad (12)$$

$$\beta_i^p = \frac{\exp(u_i^p)}{\sum_{k=1}^{N_\rho} \exp(u_k^p)} \quad (13)$$

$$\tilde{r} = \sum_{i=1}^{N_\rho} \beta_i^p p_i \quad (14)$$

### 3.7 谣言分类模块

在谣言分类模块中, 我们将得到的谣言传播树表示  $\tilde{r}$  输入到一层前馈神经网络(Feed-forward layer)和 softmax 层, 计算得到谣言的预测标签  $\tilde{y}$ :

$$\tilde{y} = \text{Softmax}(W \tilde{r} + b) \quad (15)$$

我们采用交叉熵作为分类损失来衡量真实标签和模型所预测标签之间的相似度。交叉熵越小, 表示预测标签越接近真实标签。因此, 对于每个谣言事件, 分类器的目标是最小化式(16)所示的交叉熵损失函数。

$$\mathcal{L}(y, \tilde{y}) = -[y \log \tilde{y} + (1-y) \log(1-\tilde{y})] + \lambda \|\theta\|_2^2 \quad (16)$$

其中,  $y$  代表事件对应的真实标签,  $\tilde{y}$  代表网络模型对事件的预测标签,  $\|\cdot\|_2^2$  代表对网络模型的所有参数  $\theta$  进行  $L_2$  正则化操作,  $\lambda$  是平衡系数。

我们使用 Adam 优化器<sup>[29]</sup> 对模型中的参数进行优化。同时, 为了抑制模型的过拟合程度, 我们在模型中加入 Dropout 正则化方法。

## 4 实验结果及分析

### 4.1 实验数据集

为了评估本文模型, 我们使用了 4 个经典的谣言数据集来对比本文模型与已有相关模型的性能: Twitter15<sup>[3]</sup>, Twitter16<sup>[3]</sup>, PHEME<sup>[30]</sup> 和 Weibo<sup>[18]</sup> 数据集。

Twitter15 和 Twitter16 数据集由 Ma 等<sup>[3]</sup> 从使用广泛的社交媒体平台 Twitter<sup>1)</sup> 上抓取整理, 并使用辟谣网站(如 snopes.com 和 Emergent.info)上的准确性标签对数据进行标签分类。Twitter15 和 Twitter16 数据集分别包含 1490 和 818 个谣言传播树结构, 每棵传播树由转发帖子和评论帖子组成。由于转发帖子不包含任何文本信息, 因此我们删去了

<sup>1)</sup> <https://twitter.com>

这些转发帖子,仅保留评论帖子。由于原始数据集没有公开评论文本,因此我们根据所公开的评论帖子 ID 使用 Twitter API<sup>1)</sup>重新爬取评论文本,并删去一些因被删除或屏蔽而失效的帖子结点,对谣言传播树结构进行重构。Twitter15 和 Twitter16 数据集包括 4 个标签:非谣言(NR)、验证为假的谣言(FR)、验证为真的谣言(TR)和未经验证的谣言(UR)。

PHEME 数据集是一个旨在查找和验证 Twitter 平台上传播的谣言的项目<sup>2)</sup>的一部分。这个数据集包含了 5 个不同的事件,一共 1972 个谣言传播树结构。PHEME 数据集由专业记者进行标注,包括 3 个标签:验证为假的谣言(False FR)、验证为真的谣言(TR)和未经验证的谣言(UR)。

Weibo 数据集由 Ma 等<sup>[18]</sup>抓取整理自微博平台,它是一个在中国应用十分广泛的社交媒体平台。Weibo 数据集一共有 4664 条谣言数据,包括两个标签:验证为假的谣言(FR)和验证为真的谣言(TR)。其中标签为 FR 的谣言通过新浪微博举报处理大厅<sup>3)</sup>抓取得到。

4 个数据集的统计信息如表 1 所列。

表 1 数据集统计信息  
Table 1 Dataset statistic

Statistic	Twitter15	Twitter16	PHEME	Weibo
# of source post	1490	818	1972	4664
# of posts	42914	20295	31430	2011057
# of true-rumors	372	205	1008	2351
# of false-rumors	370	205	393	2313
# of unverified-rumors	374	203	571	0
# of non-rumors	374	205	0	0
Avg. # of depth/tree	2.61	2.55	4.12	2.85
Avg. # of paths/tree	26.16	22.53	10.24	394.11
Avg. # of posts/tree	28.80	24.81	15.94	431.27
Avg. # of length/root post	17.68	17.04	16.44	59.24
Avg. # of length/post	14.13	13.68	16.12	9.59

## 4.2 基线模型

我们对比了本文模型 DAN-Tree 与已有经典模型之间的性能差异,包括基于特征工程的传统分类方法(DTR<sup>[12]</sup>, DTC<sup>[13]</sup>, RFC<sup>[14]</sup>, SVM<sup>[15-16]</sup>)、基于时间序列建模的深度学习方法(GRU<sup>[18]</sup>, PPC<sup>[10]</sup>, PLAN<sup>[11]</sup>)、已有的传播树结构学习方法(PTK<sup>[3]</sup>, RvNN<sup>[5,24]</sup>, Sta-PLAN<sup>[11]</sup>, Bi-GCN<sup>[6]</sup>)等。对比的基线模型的简要描述如下:

DTR 模型<sup>[12]</sup>。该模型是一个基于决策树的排名模型,通过搜索查询短语来识别趋势谣言。

DTC 模型<sup>[13]</sup>。该模型是一个基于决策树的模型,它使用帖子中人工设计的统计特征来训练决策树用于分类。

RFC 模型<sup>[14]</sup>。该模型是一个随机森林分类器,它使用人工选择的用户、文本和结构等特征来训练分类器。

SVM-RBF 模型<sup>[16]</sup>。该模型是一个带有径向基函数(Radial Basis Function, RBF)核的 SVM 分类器,同样使用了一系列人工设计的统计特征。

SVM-TS 模型<sup>[15]</sup>。该模型是一种对时间序列上特征值随时间的变化进行建模的 SVM 分类器。

GRU 模型<sup>[18]</sup>。该模型是一种基于 GRU 循环神经网络

的深度学习模型,它通过对时间序列上的用户评论进行建模来学习谣言表示并进行分类预测。

PPC 模型<sup>[10]</sup>。该模型在时间序列上联合使用 RNN 和 CNN 模型,通过谣言传播路径上的用户特征得到谣言表示。但该模型没有使用谣言传播过程中帖子结点的语义信息和结构信息。

PLAN 模型<sup>[11]</sup>。该模型是一个基于 Transformer 结构的谣言表示学习模型,它将谣言传播过程建模为时间序列,并利用 Transformer 注意力机制学习谣言的表示,可以学习到序列化的帖子间较长距离的依赖关系。但该模型同样忽略了对结构信息的利用。

PTK 模型<sup>[3]</sup>。该模型是一种带有传播树核的 SVM 分类器,它通过核方法捕获传播树结构之间的相似性,从而进行谣言分类。

RvNN 模型<sup>[5]</sup>。该模型使用 GRU 模块对谣言传播树结构进行自顶向下和自底向上的递归建模,通过聚合文本信息和结构信息得到谣言表示并进行谣言类别预测。由于自顶向下的建模方法效果优于自底向上的建模方法,因此,本文选用自顶向下的建模方法进行模型对比。

RvNN\* 模型<sup>4)</sup>。该模型是 RvNN 模型的一个改进版本。在训练时的优化器选择中,该模型使用 AdaGrad 算法<sup>[31]</sup>替换动量梯度下降算法。

RvNN-GA 模型<sup>[24]</sup>。该模型是 RvNN 模型的一个变体。它对经过 GRU 模块进行递归建模之后的所有结点使用全局注意力方法,通过加权求和得到谣言传播树的表示来进行类别预测。同理,本文选用自顶向下的建模方法进行模型对比。

Sta-PLAN 模型<sup>[11]</sup>。该模型是 PLAN 模型的一个变体。它在 PLAN 模型的基础上添加了一个用于描述帖子间反复关系的变量,在一定程度上考虑了谣言的传播结构信息。

Bi-GCN 模型<sup>[6]</sup>。该模型是一个基于图卷积(GCN)的谣言传播树表示学习模型,它从自顶向下和自底向上两个方向对谣言的传播结构进行联合建模,以模拟谣言的传播和扩散过程,具有较强的传播树结构表示学习能力。

RvNN 和 RvNN\* 模型没有在 Weibo 和 PHEME 数据集上进行验证,PLAN 模型没有在 Weibo 数据集上进行验证, Bi-GCN 模型没有在 PHEME 数据集上进行验证。我们使用这些模型公开的代码,完善这些缺失的任务。

## 4.3 数据预处理

类似于 RvNN<sup>[5]</sup>, Bi-GCN<sup>[6]</sup> 和 RvNN-GA<sup>[24]</sup> 等方法,我们在 4 个数据集上使用 5 折交叉验证的方法来训练和测试模型效果。

本文使用的 Twitter15, Twitter16 和 PHEME 数据集是从英文社交平台 Twitter 上抓取的,因此我们使用 NLTK (Natural Language Toolkit)库<sup>5)</sup>中的 WordNetLemmatizer 方法进行词形还原,并使用 TweetTokenizer 方法对文本进行分词操作。

对于从中文社交平台微博上抓取的 Weibo 数据集,我们

<sup>1)</sup> <https://dev.twitter.com/rest/public>

<sup>2)</sup> <https://www.pHEME.eu/>

<sup>3)</sup> <https://service.account.weibo.com/>

<sup>4)</sup> [https://github.com/majingCUHK/Rumor\\_RvNN/pull/7](https://github.com/majingCUHK/Rumor_RvNN/pull/7)

<sup>5)</sup> <https://www.nltk.org/>

去除了数据中不包含语义信息的超链接,最后使用 jieba 分词<sup>1)</sup>对数据进行分词操作。

对这 4 个数据集,我们没有删除“!”“?”等标点符号,并且在分词操作中保留了文本中可能出现的 emoji 及其他表情符号,如“:),:(“。这些符号具有一定的情感表达能力,有助于在谣言检测任务中获得更合适的句子级语义表示。最后,我们选用数据集中出现频率最高的前 5000 个词进行模型训练。

#### 4.4 评估指标和参数设置

为了更公平地比较各模型的实验结果,本文和之前的工作一样,在 Twitter15, Twitter16 和 Weibo 数据集上使用正确率(Accuracy, Acc.)作为模型效果的评价指标。正确率越高,模型的性能越好。对于 PHEME 这种各类数据分布不平衡的数据集,我们主要使用  $F_1$  值( $F_1$  score)来评价各模型性能的优劣。 $F_1$  值越高,模型的综合性能越好。在 4 个数据集的子类上,我们同样使用  $F_1$  值指标。在 Weibo 数据集上,我们还使用精确率(Precision, Prec.)和召回率(Recall, Rec.)来评价各个子类上的实验效果。

实验环境使用 pytorch<sup>2)</sup>, Adam 算法中  $\beta_1$  和  $\beta_2$  的值分别设置为 0.9 和 0.999, 学习率  $lr$  的初始值设置为 0.01, dropout 概率设置为 0.5, 平衡系数  $\lambda$  设置为 0.01, 隐层维度设置为 300。模型中的词嵌入被初始化为 300 维的词向量,并且在模型训练过程中保持词向量为可训练模式。定长传播路径序列的长度  $N$  设置为 50,  $N_i$  设置为 1 层,多头注意力中的  $h$  设置为 4, Twitter15, Twitter16 和 PHEME 数据集采用相同的实验参数。由于 Weibo 数据集中平均帖子数量远多于其他 3 个数据集,并且拥有较大的平均传播路径数量,因此我们在 Weibo 数据集上的实验中,将多头注意力中的  $h$  设置为 6,  $N$  设置为 90。Weibo 数据集的其他参数设置同其他 3 个数据集一致。

#### 4.5 实验结果

表 2—表 5 列出了本文模型 DAN-Tree 和其他基线方法在 Twitter15, Twitter16, PHEME 和 Weibo 4 个数据集上的谣言检测效果。表中每个评价指标的最优值用加粗的字体显示。

表 2 Twitter15 数据集上的实验结果

Table 2 Experimental results on Twitter15

Model	Acc.	NR	FR	TR	UR
		$F_1$	$F_1$	$F_1$	$F_1$
DTR	0.409	0.501	0.311	0.364	0.473
DTC	0.454	0.733	0.355	0.317	0.415
RFC	0.565	0.810	0.422	0.401	0.543
SVM-RBF	0.318	0.455	0.037	0.218	0.225
SVM-TS	0.544	0.796	0.472	0.404	0.483
GRU	0.641	0.684	0.634	0.688	0.571
PPC	0.842	0.811	0.875	0.818	0.790
PLAN	0.845	0.823	0.858	0.895	0.802
PTK	0.667	0.619	0.669	0.772	0.645
RvNN	0.723	0.682	0.758	0.821	0.654
RvNN*	0.778	0.742	0.809	0.804	0.758
RvNN-GA	0.756	0.784	0.774	0.817	0.680
Sta-PLAN	0.852	0.840	0.846	0.884	0.837
Bi-GCN	0.886	0.891	0.860	0.930	0.864
DAN-Tree	<b>0.902</b>	<b>0.891</b>	<b>0.900</b>	<b>0.930</b>	<b>0.886</b>

表 3 Twitter16 数据集上的实验结果

Table 3 Experimental results on Twitter16

Model	Acc.	NR	FR	TR	UR
		$F_1$	$F_1$	$F_1$	$F_1$
DTR	0.414	0.394	0.273	0.630	0.344
DTC	0.465	0.643	0.393	0.419	0.403
RFC	0.585	0.752	0.415	0.547	0.563
SVM-RBF	0.321	0.423	0.085	0.419	0.037
SVM-TS	0.574	0.755	0.420	0.571	0.526
GRU	0.633	0.617	0.715	0.577	0.527
PPC	0.863	0.820	<b>0.898</b>	0.843	0.837
PLAN	0.874	0.853	0.839	0.917	0.888
PTK	0.662	0.643	0.623	0.783	0.655
RvNN	0.737	0.662	0.743	0.835	0.708
RvNN*	0.788	0.763	0.778	0.853	0.761
RvNN-GA	0.764	0.708	0.753	0.840	0.738
Sta-PLAN	0.868	0.826	0.833	0.927	0.888
Bi-GCN	0.880	0.847	0.869	0.937	0.865
DAN-Tree	<b>0.901</b>	<b>0.877</b>	0.865	<b>0.953</b>	<b>0.908</b>

表 4 PHEME 数据集上的实验结果

Table 4 Experimental results on PHEME

Model	Acc.	Macro- $F_1$	FR	TR	UR
			$F_1$	$F_1$	$F_1$
RvNN	0.728	0.749	0.761	0.717	0.769
RvNN*	0.743	0.758	0.770	0.745	0.759
PLAN	0.785	0.772	0.753	0.828	0.735
Bi-GCN	0.722	0.677	0.570	0.792	0.675
DAN-Tree	<b>0.845</b>	<b>0.830</b>	<b>0.792</b>	<b>0.874</b>	<b>0.823</b>

表 5 Weibo 数据集上的实验结果

Table 5 Experimental results on Weibo

Model	Class	Acc.	Prec.	Rec.	$F_1$
DTR	FR	0.789	0.784	0.801	0.793
	TR		0.794	0.777	0.785
DTC	FR	0.831	0.847	0.815	0.831
	TR		0.815	0.824	0.819
RFC	FR	0.855	0.810	0.929	0.866
	TR		0.916	0.779	0.842
SVM-RBF	FR	0.879	0.777	0.656	0.708
	TR		0.579	0.708	0.615
SVM-TS	FR	0.885	0.950	0.932	0.938
	TR		0.124	0.047	0.059
PTK	FR	0.891	0.876	0.913	0.894
	TR		0.907	0.868	0.887
RvNN	FR	0.908	0.912	0.897	0.905
	TR		0.904	0.918	0.911
RvNN*	FR	0.929	0.949	0.909	0.928
	TR		0.911	0.950	0.930
PPC	FR	0.921	0.896	0.962	0.923
	TR		0.949	0.889	0.918
PLAN	FR	0.943	0.939	0.948	0.943
	TR		0.946	0.937	0.942
Bi-GCN	FR	<b>0.961</b>	<b>0.961</b>	0.964	<b>0.961</b>
	TR		0.962	<b>0.962</b>	<b>0.960</b>
DAN-Tree	FR	0.958	0.946	<b>0.972</b>	0.958
	TR		<b>0.972</b>	0.945	0.958

根据表 2—表 5 的实验结果可知, DAN-Tree 模型在 Twitter15 和 Twitter16 数据集上的正确率分别达到了 90.2% 和 90.1%, 在 PHEME 数据集上的正确率和  $F_1$  值分别达到了 84.5% 和 83.0%, 在 Weibo 数据集上的正确率达到了 95.8%。DAN-Tree 模型在 Twitter15, Twitter16 和 PHEME 数据集上的实验效果达到了最优, 在 Weibo 数据集

<sup>1)</sup> <https://github.com/fxsjy/jieba>

<sup>2)</sup> <https://pytorch.org/>

上也达到了同最优结果相当的效果,表明了 DAN-Tree 模型在谣言检测领域中相比其他基线方法具有很大优势。

从表 2、表 3 和表 5 的实验结果来看,使用特征工程的传统方法(DTR, DTC, RFC, SVM-RBF 和 SVM-TS)在 Twitter15, Twitter16 和 Weibo 数据集上的效果不够理想,这说明了传统方法缺乏从谣言数据中提取高层表示的能力。SVM-TS 和 RFC 模型在这些传统方法中表现出了最好的效果,可能的原因是这两种方法使用了谣言事件的时间和结构特征。

在对谣言的时间序列特征进行建模的深度学习方法中,可以发现,PPC 模型由于联合使用了 RNN 和 CNN 结构而表现出了比 GRU 模型更好的检测效果;PLAN 模型使用 Transformer 结构作为特征抽取器并关注到了帖子之间的隐式关系,因而在检测效果上得到了进一步的提升。Sta-PLAN 在 PLAN 模型的基础上添加了一个用来描述帖子间回复类型的变量,粗略地利用了谣言的结构信息,在 Twitter15 数据集上取得了更好的谣言检测效果。

在使用了谣言结构信息的方法中,PTK 模型使用了一个具有传播树核的 SVM 分类器来度量传播树之间的相似性。而使用了传播树结构的递归神经网络表示学习模型 RvNN,具有学习传播树结构深层语义的能力,在实验效果上相比 PTK 模型有了较大的提升。RvNN-GA 模型在 RvNN 的基础上使用全局注意力机制关注不同帖子结点对传播树表示的不同重要度,达到了更好的效果。Bi-GCN 模型利用双向图卷积结构关注谣言的传播和扩散过程,在 3 个数据集上取得了基线方法中的最好效果。

本文提出的谣言传播树结构上的双注意力模型 DAN-Tree 使用 Transformer 结构学习传播路径中帖子的隐式语义关系,并利用注意力机制学习传播树路径上的帖子结点注意力和传播路径注意力,较好地捕获了传播树结构上的语义信息。此外,路径过采样技术和结构嵌入方法使得模型可以更好地学习谣言传播树的深层结构信息。相比目前已有工作的最好效果,DAN-Tree 模型在 Twitter15 和 Twitter16 数据集上的正确率由 88.6%和 88.0%提升到了 90.2%和 90.1%,分别提高了 1.81%和 2.39%,在 PHEME 数据集上的  $F_1$  值由 77.2%提升到了 83.0%,提高了 7.51%,均优于所有的对比模型。此外,DAN-Tree 模型在这 3 个数据集的各个类别的  $F_1$  值也几乎全部达到了最优值。

DAN-Tree 模型在 Weibo 数据集上没能达到最优结果。分析其中原因,根据表 1 的统计信息,即使考虑到源帖子长度和标点符号的数量,Weibo 数据集中的帖子平均也仅有 9.59 个中文字符,DAN-Tree 模型没能提取到帖子合适的语义特征,无法有效利用 Transformer 结构提取高层语义特征。但是 DAN-Tree 模型在 Weibo 数据集上的正确率仍达到了 95.8%,呈现出和最优结果相当的效果。此外,相比使用了 RNN 模块建模传播树结构的 RvNN 模型和提取了高层语义特征但缺少结构信息的 PLAN 模型,DAN-Tree 模型实现了对谣言传播树的深层语义信息和结构信息的有效融合,在 Weibo 数据集上的检测效果有了很大提升。

#### 4.6 消融实验

为验证 3.3 节中结构嵌入(Structure Embedding, SE)对

谣言检测效果的增益作用,我们固定其他实验参数不变,观察分析“采用”或“移除”结构嵌入的模型在 4 个数据集上的实验效果变化,结果如表 6—表 9 所列。其中,“DAN-Tree”代表“采用”结构嵌入的模型,“w/o SE”代表“移除”结构嵌入的模型。

表 6 Twitter15 数据集上的消融实验结果

Table 6 Ablation Experimental results on Twitter15

Model	Acc.	NR	FR	TR	UR
		$F_1$	$F_1$	$F_1$	$F_1$
DAN-Tree	0.902	0.891	0.900	0.930	0.886
w/o SE	0.895	0.895	0.896	0.918	0.871

表 7 Twitter16 数据集上的消融实验结果

Table 7 Ablation Experimental results on Twitter16

Model	Acc.	NR	FR	TR	UR
		$F_1$	$F_1$	$F_1$	$F_1$
DAN-Tree	0.901	0.877	0.865	0.953	0.908
w/o SE	0.898	0.862	0.869	0.945	0.918

表 8 PHEME 数据集上的消融实验结果

Table 8 Ablation Experimental results on PHEME

Model	Acc.	Macro- $F_1$	FR	TR	UR
			$F_1$	$F_1$	$F_1$
DAN-Tree	0.845	0.830	0.792	0.874	0.823
w/o SE	0.836	0.819	0.781	0.870	0.805

表 9 Weibo 数据集上的消融实验结果

Table 9 Ablation Experimental results on Weibo

Model	Class	Acc.	Prec.	Rec.	$F_1$
DAN-Tree	FR	0.958	0.946	0.972	0.958
	TR		0.972	0.945	0.958
w/o SE	FR	0.952	0.949	0.954	0.952
	TR		0.955	0.950	0.952

从表 6—表 9 的实验结果来看,“采用”结构嵌入的模型在谣言检测任务上的效果均优于“移除”结构嵌入的模型。同时,即使是“移除”结构嵌入的模型,依然在 Twitter15, Twitter16 和 PHEME 数据集上达到了优于其他所有基线方法的效果。在传播树的平均深度较浅的 Twitter16 数据集上,结构嵌入方法在正确率指标上仅提高了 0.3%。但是对于传播树深度较深的 Twitter15, Weibo 和 PHEME 数据集,结构嵌入方法在它们各自的评价指标上分别提高了 0.7%, 0.7% 和 1.1%。这说明结构嵌入方法对传播树更深的的数据有着更好的增益效果。

据此我们可以得出结论:结构嵌入方法为模型补充了帖子结点在谣言传播树中重要的空间位置信息,弥补了 Transformer 结构对位置不敏感的缺陷,有效利用了谣言传播过程中的传播结构信息,从而提升了谣言检测任务的效果。

#### 4.7 参数分析

我们在 DAN-Tree 模型中使用了固定大小的帖子文本长度和传播路径数量  $N_p$ 。这些超参的选择可能会影响模型的性能。本节将进行一系列的参数敏感性实验,研究这些超参对模型性能的影响。在实验中,我们使用控制变量法来减少无关参数的影响。

图 3 给出了在不同的帖子文本长度下,DAN-Tree 模型在 4 个数据集上的性能变化情况。从图中的实验结果可以看出,帖子的文本长度对模型的性能有显著影响。帖子的长度

过短时,模型无法学习到足够的语义信息,导致模型在4个数据集上的效果均显著下降。这说明帖子长度的增加有助于模型检测性能的提升。结合表1中的统计信息,我们发现,随着帖子长度逐渐增加到数据集中源帖子的平均单词数量,模型的性能逐渐提升并最终趋于稳定。说明源帖子的语义信息可以更加有效地提高模型在谣言检测任务上的效果。

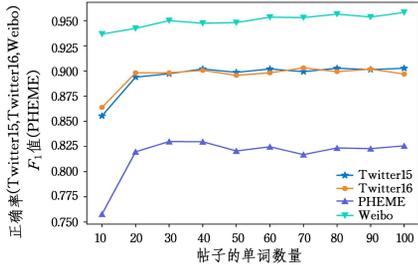


图3 帖子的文本长度对实验效果的影响

Fig. 3 Influence of post text length on experimental performance

图4给出了谣言传播树的路径数量对模型检测效果的影响。传播路径中的评论帖子可以为谣言事件真实性的检测提供指向性的线索,通常认为,深度更深的传播路径在帖子之间进行“互动”的过程中可以为谣言检测任务提供更可信的信息。

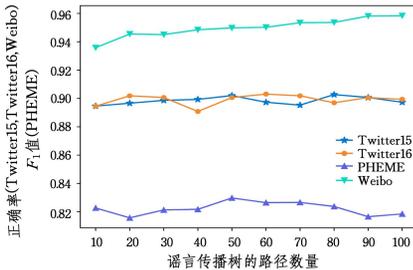


图4 传播树的路径数量对实验效果的影响

Fig. 4 Influence of path number of propagation tree on experimental performance

结合表1中的统计信息和图4中的实验结果,我们发现,随着传播路径数量 $N_p$ 逐渐增加至数据集中各传播树所包含的平均传播路径数量,模型的检测效果逐渐提升。当 $N_p$ 值逐渐超过数据集的平均传播路径数量后,模型在Twitter15, Twitter16和PHEME数据集上的检测效果出现了一定范围内的波动。我们认为这是由于在3.4节算法1所描述的路径过采样技术中,随机过采样传播路径虽然具备提高模型检测效果的能力,但某些情况下可能会引入一些噪声。 $N_p$ 值达到90时,模型在Weibo数据集上的检测效果已趋于稳定,此时 $N_p$ 值尚未达到Weibo数据集的平均传播路径数量。我们认为这是由于 $N_p$ 值达到90时,对于Weibo数据集来说,传播路径中已经包含了丰富的讨论信息,因此可以达到较好且稳定的检测效果。其中,模型在PHEME数据集上取得最优效果时所需的传播路径数量为50,超过了该数据集的平均路径数量。我们认为这是由于该数据集的传播树较深,在路径过采样技术中使得更深的传播路径具有更大的概率被多次学习,使得模型充分学习帖子之间的指向性线索。综上,我们认为数量更多、深度更深的传播路径将提供更多有意义的信息,有助于模型在谣言检测任务中的学习。

此外,DAN-Tree模型在Twitter15, Twitter16和PHEME数据集上,仅使用数量为10的传播路径就取得了优于其他基线方法的效果,体现了DAN-Tree模型在谣言检测任务上的优越性。

**结束语** 本文提出了一种基于传播树结构的结点和路径双注意力谣言检测模型DAN-Tree。该模型使用Transformer编码结构中的注意力机制学习传播路径中用户之间的互动关系,并在帖子注意力学习的基础上,学习不同路径对谣言检测的重要性,实现了传播树结构上的深层结构和语义信息的有机融合。此外,我们还使用了路径过采样技术及结构嵌入的方法来增强学习传播树结构上的深层结构信息。在4个经典的谣言检测领域数据集中,本文提出的模型在其中的3个数据集上均优于其他基线方法,充分证明了在传播结构中考虑帖子间的隐式关系对谣言检测效果的重要性。考虑到除了各帖子文本之间存在的回复关系,帖子和发表帖子的用户之间也存在着链接关系,可信的用户特征及其与帖子关系的学习对谣言的检测有一定的效果,在未来的研究中,我们计划将谣言传播过程中的用户特征及其与帖子之间的关系添加到模型的学习过程之中,给出更有效的谣言检测方法。另外,现有的有标定的谣言数据集规模不大,而谣言数据又常存在一经认证则链接失效的问题,收集一个大规模的、具有深层传播结构的谣言数据集对谣言检测研究具有重要的意义。因此,长期跟踪并收集整理一个大规模的、特征全面的谣言检测数据集也是我们关注的重点。

## 参考文献

- [1] CHEN Y F, LI Z Y, LIANG X, et al. Review on rumor detection of online social networks[J]. Chinese Journal of Computers, 2018, 41(7): 1648-1676.
- [2] ZUBIAGA A, AKER A, BONTCHEVA K, et al. Detection and resolution of rumours in social media: a survey[J]. ACM Computing Surveys, 2018, 51(2): 1-36.
- [3] MA J, GAO W, WONG K F. Detect rumors in microblog posts using propagation structure via kernel learning[C] // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL, 2017: 708-717.
- [4] WU K, YANG S, ZHU K Q. False rumors detection on sina weibo by propagation structures[C] // Proceedings of the 31st IEEE International Conference on Data Engineering. New York: IEEE, 2015: 651-662.
- [5] MA J, GAO W, WONG K F. Rumor detection on twitter with tree-structured recursive neural networks[C] // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2018: 1980-1989.
- [6] BIAN T, XIAO X, XU T Y, et al. Rumor detection on social media with bi-directional graph convolutional networks[C] // Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence. Menlo Park, CA: AAAI, 2020: 549-556.
- [7] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C] // Proceedings of the 31st International Conference on Neural Information Processing Systems. New York: Curran Associates, 2017: 6000-6010.

- [8] NGUYEN X P, JOTY S, HOI A, et al. Tree-structured attention with hierarchical accumulation[C]// Proceedings of the 8th International Conference on Learning Representations. OpenReview.net, 2020.
- [9] DAI Z H, YANG Z L, YANG Y M, et al. Transformer-xl: attentive language models beyond a fixed-length context[C]// Proceedings of the 57th Conference of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2019: 2978-2988.
- [10] RAFFEL C, SHAZEER N, ROBERTS A, et al. Exploring the limits of transfer learning with a unified text-to-text transformer[J]. Journal of Machine Learning Research, 2020, 21: 1-67.
- [11] KHOO L M S, CHIEU H L, QIAN Z, et al. Interpretable rumor detection in microblogs by attending to user interactions[C]// Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence. Menlo Park, CA: AAAI, 2020: 8783-8790.
- [12] ZHAO Z, RESNICK P, MEI Q Z. Enquiring minds: early detection of rumors in social media from enquiry posts[C]// Proceedings of the 24th International Conference on World Wide Web. New York: ACM, 2015: 1395-1405.
- [13] CASTILLO C, MENDOZA M, POBLETE B. Information credibility on twitter[C]// Proceedings of the 20th International Conference on World Wide Web. New York: ACM, 2011: 675-684.
- [14] KWON S, CHA M, JUNG K. Prominent features of rumor propagation in online social media[C]// Proceedings of the 13th International Conference on Data Mining. New York: IEEE, 2013: 1103-1108.
- [15] MA J, GAO W, WEI Z Y, et al. Detect rumors using time series of social context information on microblogging websites[C]// Proceedings of the 24th ACM International Conference on Information and Knowledge Management. New York: ACM, 2015: 1751-1754.
- [16] YANG F, YU X H, LIU Y, et al. Automatic detection of rumor on sina weibo[C]// Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics. New York: ACM, 2012: 1-7.
- [17] LIAO X W, HUANG Z, YANG D D, et al. Rumor detection in social media based on a hierarchical attention network[J]. SCIENTIA SINICA Informationis, 2018, 48: 1558-1574.
- [18] MA J, GAO W, MITRA P, et al. Detecting rumors from microblogs with recurrent neural networks[C]// Proceedings of the 25th International Joint Conference on Artificial Intelligence. Menlo Park, CA: AAAI, 2016: 3818-3824.
- [19] YU F, LIU Q, WU S, et al. A convolutional approach for misinformation identification[C]// Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence. ijcai.org, 2017: 3901-3907.
- [20] LIU Y, WU Y F B. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks[C]// Proceedings of the Thirty-second AAAI Conference on Artificial Intelligence. Palo Alto, CA: AAAI, 2018: 354-361.
- [21] GUO H, CAO J, ZHANG Y Z, et al. Rumor detection with hierarchical social attention network[C]// Proceedings of the 27th ACM International Conference on Information and Knowledge Management. New York: ACM, 2018: 943-951.
- [22] YUAN C Y, MA Q W, ZHOU W, et al. Jointly embedding the local and global relations of heterogeneous graph for rumor detection[C]// Proceedings of the 2019 IEEE International Conference on Data Mining(ICDM). New York: IEEE, 2019: 796-805.
- [23] KUMAR S, CARLEY K M. Tree lstms with convolution units to predict stance and rumor veracity in social media conversations[C]// Proceedings of the 57th Conference of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2019: 5047-5058.
- [24] MA J, GAO W, JOTY S R, et al. An attention-based rumor detection model with tree-structured recursive neural networks[J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2020, 11(4): 1-28.
- [25] MA J, GAO W. Debunking rumors on twitter with tree transformer[C]// Proceedings of the 28th International Conference on Computational Linguistics. ICCL, 2020: 5455-5466.
- [26] DEVLIN J, CHANG M W, LEE K, et al. Bert: pre-training of deep bidirectional transformers for language understanding[C]// Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, PA: ACL, 2019: 4171-4186.
- [27] YANG L, WANG P, LI H, et al. A holistic representation guided attention network for scene text recognition[J]. Neurocomputing, 2020, 414: 67-75.
- [28] YU D L, LI X, ZHANG C Q, et al. Towards accurate scene text recognition with semantic reasoning networks[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2020: 12110-12119.
- [29] KINGMA D P, BA J. Adam: a method for stochastic optimization[C]// Proceedings of the 3rd International Conference on Learning Representations. 2015.
- [30] ZUBIAGA A, LIAKATA M, PROCTER R, et al. Analysing how people orient to and spread rumours in social media by looking at conversational threads[J]. PloS one, 2016, 11(3): e0150989.
- [31] DUCHI J, HAZAN E, SINGER Y. Adaptive subgradient methods for online learning and stochastic optimization[J]. Journal of Machine Learning Research, 2011, 12(7): 2121-2159.



**HAN Xueming**, born in 1998, postgraduate. His main research interests include natural language processing and rumor detection.



**JIA Caiyan**, born in 1976, Ph.D, professor. Her main research interests include data mining, social computing and natural language processing.