计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210600046

基于改进位置编码的谣言检测模型

姜梦函　李邵梅　郑洪浩　张建朋
中国人民解放军战略支援部队信息工程大学信息技术研究所　郑州 450000
(13513127249@163.com)

摘　要　随着在线社交网络的兴起,人们传播和获取信息的方式发生了翻天覆地的变化。社交媒体在方便人们生活的同时,也加速了谣言的产生和传播。因此,如何准确高效地检测谣言成为了亟待解决的问题。为了提高谣言检测的精度,对基于全局-局部注意网络的谣言检测模型进行了改进,考虑到文本中词与词之间的位置关系对谣言检测的影响,引入了一种新的相对位置编码方法来改进原有模型的局部特征提取模块。该方法能够更准确地提取谣言中文本的语义信息和位置信息并将它们聚合,得到更优的区分谣言与非谣言的文本特征,将该特征和描述转发行为的全局特征相结合,进而提升对谣言的检测效果。实验结果表明,与其他主流检测方法相比,所提方法在微博数据集上的 $F1$ 值可达 95.0%,具有更好的检测效果。

关键词:谣言检测;深度学习;注意力机制;相对位置编码;谣言文本特征
中图法分类号　TP391

Rumor Detection Model Based on Improved Position Embedding

JIANG Meng-han,LI Shao-mei,ZHENG Hong-hao and ZHANG Jian-peng
Institute of Information Technology,PLA Strategic Support Force Information Engineering University, Zhengzhou 450000,China

Abstract　With the rise of online social networks, the way people disseminate and obtain information has changed drastically. While social media facilitates people's lives, it also accelerates the generation and spread of rumors. For this reason, detect rumors accurately and efficiently becomes an urgent problem to be solved. In order to improve the accuracy of rumor detection, the rumor detection model based on the global-local attention network has been improved. Taking into account the influence of the positional relationship between words in the text on rumor detection, a new relative position encoding method is introduced to improve the local feature extraction module of the original model. This method can more accurately extract and aggregate the semantic information and location information of the text in the rumor, and obtain better text features that distinguish between rumors and non-rumors. The combination of features and global features describing forwarding behavior improves the detection effect of rumors. Experimental results show that, compared with other mainstream detection methods, the $F1$ value of the proposed method can reach 95.0% on the Microblog data set, which has a better detection effect.

Keywords　Rumor detection, Deep learning, Attention mechanism, Relative position embedding, Rumor text features

1　引言

谣言是在人与人之间传播的,含有公众关心的信息,且真实性不能很快得到证明或者得不到证明的一种特殊陈述[1],主要涉及突发事件、公共领域、政治人物、颠覆传统、离经叛道等内容。

近年来,随着在线社交媒体的兴起,社交网站如人人网、新浪微博、微信、Facebook 以及 Twitter 等平台的蓬勃发展,人们获取信息的方式发生了翻天覆地的变化[2]。2020 年 9 月中国互联网络信息中心(CNNIC)发布的第 47 次中国互联网络发展状况报告[3]表明,截至 2020 年 12 月,我国网民规模达 9.89 亿,互联网普及率达 70.4%。社交媒体在方便人们生活的同时,也加速了谣言的产生和传播。比如,2011 年日本福岛核电站爆炸事件,有人称"食用碘盐可防辐射",随后该条谣言引发了人们大量的转发、传播,从而导致了一场全国范围内的辐射恐慌和抢盐风波,对整个社会造成极大的负面影响。

根据新浪微博辟谣官微 2021 年 2 月 7 日发布的"2020 年度微博辟谣数据报告"显示,2020 年微博站方共有效处理不实信息 76 107 条,新增谣言案例 782 例。这些信息在未经证实的情况下被迅速地歪曲和放大,误导了公众。谣言无节制地在网络上传播不仅会影响社会的和谐稳定,甚至会威胁

到稿日期:2021-06-07　**返修日期**:2021-10-20
基金项目:国家自然基金青年科学基金(62002384);郑州市协同创新重大专项(162/32410218);中国博士后科学基金面上项目(47698)
This work was supported by the Young Scientists Funds of the National Natural Science Foundation of China(62002384), Zhengzhou Collaborative Innovation Major Project(162/32410218) and China Postdoctoral Science Foundation(47698).
通信作者:李邵梅(m19139795259@163.com)

国家或地区安全。但由于人工检测网络谣言耗时耗力，且设计出来的特征往往局限于特定场景，泛化能力不好，因此，如何准确高效地检测出社交网络谣言成为一个重要的研究方向。

谣言检测通常被归为二分类问题，现有的检测方法大致可分为两类。

(1)基于传统机器学习的方法。该类方法将整个文本分类问题拆分成特征工程和分类器建模两部分，特征工程分为文本预处理、特征提取、文本表示3部分，其最终目的是将文本表示为计算机可以识别的、能够代表文本特征的特征矩阵。在将文本表示为模型可以处理的向量数据后，即可使用机器学习模型进行分类，常用的模型有朴素贝叶斯、K-最近邻法、决策树、支持向量机(Support Vector Machines，SVM)等。Castillo等[4]从Twitter数据集中提取情感分数、包含URL的微博数、用户注册天数等特征，并利用决策树算法来检测谣言；Yang等[5]从新浪微博数据集中提取用户的地理位置、发布微博的客户端以及文本符号的情感极性等特征，利用SVM分类器进行谣言检测；Kwon等[6]从时间特征、结构特征和语言特征3个方面来改善谣言检测的效果，并在SVM、决策树和随机森林3种算法上进行了对比实验。一条消息除了原文内容外，还包括转发数、评论数等特征量，它的所有转发路径可以形成树状结构，这种结构通常被称为传播树。树的根节点表示原文用户，其他节点表示转发用户，而节点间连接与否表明用户间是否存在转发关系。Wu等[7]把每个消息的传播过程构建成传播树，并结合基于内容、用户和传播特征的径向基核函数构成混合SVM分类器进行谣言识别，取得了较好的结果。Ma等[8]从传播模式入手，提出了一种基于传播树核(Propagation Tree Kernel)的方法，该方法根据谣言的传播结构构建传播树，并通过评估传播树结构之间的相似性来捕获区分不同类型谣言的高阶模式，最后使用树为核函数的SVM分类器进行分类。但是，传统的机器学习方法需要人工提取特征，耗时耗力，成本太高，泛化能力不好。

(2)基于深度学习的端到端的方法。2012年以来，随着深度学习技术的发展及其在自然语言处理领域应用的深入，研究者提出采用深度学习方法来自动提取特征并检测谣言。基于深度学习方法最主要的优势是能够利用循环神经网络和卷积神经网络等深度学习模型自动获取谣言更深层次的特征。Ma等[9]于2016年首次将深度学习模型应用于社交网络谣言检测任务，利用RNN模型自动学习相关帖子的文本表示，挖掘数据的深层特征。近期有研究使用深度学习方法从传播路径中挖掘出高阶的表示，以进行谣言的识别。2018年，Ma等[10]基于递归神经网络对传播树进行建模，构造了自底向上和自顶向下的树形结构，该网络可以捕获谣言随时间传播的序列特征，用于Twitter数据集上的谣言检测，相比文献[8]中的方法，这个方向不需要比较。2020年，Ma等[11]提出一种基于树形Transformer的检测模型，利用3种Transformer的变体架构构建传播树，将一条消息可能触发的一系列回复定义为子树，多个子树构成整个树的层次结构，并根据传播树的结构深入挖掘用户的观点并优化不准确的信息。在TWITTER和PHEME数据集上的实验结果表明，该方法能不断提升谣言检测性能。鉴于图卷积网络(Graph Convolutional Networks，GCN)在社交网络、物理系统、化学制药发现等领域取得的成就，Bian等[12]从谣言自顶向下和自底向上两个传播方向挖掘特征，通过双向图卷积网络捕获全局结构信息，第一次将GCN用于社交媒体的谣言检测。Yuan等[13]提出了全局-局部注意力网络(Global-Local Attention Network，GLAN)，该网络结构是将局部语义信息和全局结构信息结合起来编码，有效提升了谣言检测的效果。

为了进一步提高基于深度学习的谣言检测效果，本文对全局-局部注意力网络进行了改进，提出了一种新的谣言检测方法。在对谣言的文本特征进行抽取时，本文引入了一种新的相对位置编码方法，能够更准确地提取谣言中文本的语义信息和位置信息，为谣言的检测提供依据，然后基于softmax分类层实现对网络谣言的检测。

2 基于全局-局部注意力网络的谣言检测模型

2.1 全局-局部注意力网络架构

Yuan等[13]提出的全局-局部注意力网络能够同时提取谣言文本的局部语义信息和全局结构信息，并将两者聚合用于谣言检测。该模型由文本表示、局部关系编码、全局关系编码、局部和全局关系谣言检测四大部分组成，结构如图1所示。

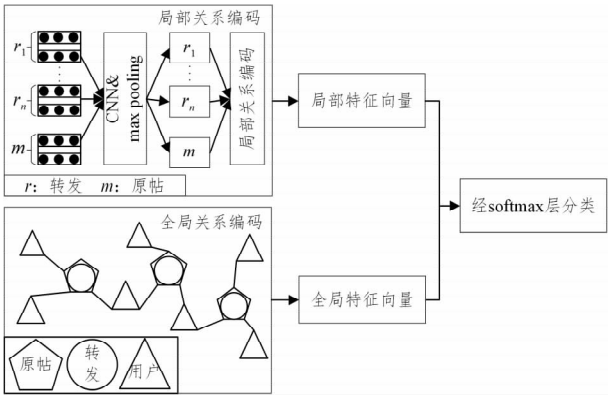


图1 GLAN模型架构

Fig. 1 GLAN model architecture

首先，文本表示模块对社交网络消息进行词嵌入，从而获得其文本表示向量；其次，局部关系编码模块利用注意力机制[14]从文本向量中挖掘更深层次的语义信息，并将其作为局部特征向量；然后，全局关系编码模块根据帖子的原始发布用户和转发用户之间的关系构建图结构，利用图注意力网络[15]捕获图中的结构信息，并将其作为全局特征向量；最后，谣言检测模块将局部特征向量和全局特征向量拼接后经softmax层进行分类。其中，局部关系编码模块使用注意力机制以更好地获取文本的语义信息。具体流程如下。

首先，输入文本$\boldsymbol{M}=[\boldsymbol{m}_1,\cdots,\boldsymbol{m}_N]$由单词嵌入$x_i$和单词的位置嵌入$\boldsymbol{u}_i$相加得到，即：

$$\boldsymbol{m}_i=\boldsymbol{x}_i+\boldsymbol{u}_i \tag{1}$$

单词嵌入$\boldsymbol{x}_i$由Word2Vec预训练得到，位置嵌入$\boldsymbol{u}_i$是

通过引入位置编码(Position Embedding, PE)的特征来捕捉序列的位置信息,计算式如下:

$$u(i,2n)=\sin\left(\frac{i}{10000^{\frac{2n}{d}}}\right)$$
$$u(i,2n+1)=\cos\left(\frac{i}{10000^{\frac{2n}{d}}}\right) \tag{2}$$

其中,i 表示单词在文本中的位置;n 表示词向量的维度;d 为输入维度,$n\in\left[0,\frac{d}{2}\right]$;$2n$ 表示偶数位置,使用正弦编码;$2n+1$ 表示奇数位置,使用余弦编码。

从式(2)可以看出,序列中不同位置的单词在某一维度上的位置编码数值也不一样,即同一序列的不同单词在某一维度符合正弦或余弦编码。如图 2 所示,当向量的维度为 128 时,取词向量的 4 个维度的图像,可以看出它们都是正弦(余弦)函数,而且周期越来越长。当词语之间的位置偏移量为 k 时,$u(i+k)$可以表示成 $u(i)$的线性函数,模型就能学习到相对位置关系。

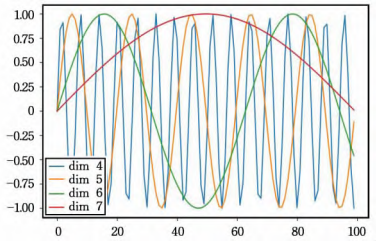


图 2 位置编码向量示例图

Fig. 2 Example diagram of position coding vector

通过线性变换将文本m_i 转化为查询向量(Query,$\boldsymbol{Q}$),键向量(Key,$\boldsymbol{K}$)和值向量(Value,$\boldsymbol{V}$):

$$\begin{aligned}\boldsymbol{Q}&=\boldsymbol{W}_q\boldsymbol{M}\\ \boldsymbol{K}&=\boldsymbol{W}_k\boldsymbol{M}\\ \boldsymbol{V}&=\boldsymbol{W}_v\boldsymbol{M}\end{aligned} \tag{3}$$

其中,$\boldsymbol{W}_q$,$\boldsymbol{W}_k$,$\boldsymbol{W}_v$ 分别为可学习的参数矩阵。

然后,利用 $\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}$ 计算自注意力(Self-Attention)的输出向量,即:

$$Attention(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=\mathrm{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{4}$$

最后,利用多头注意力(Multi-Head Attention)对式(4)的结果进行拼接,作为注意力机制的最终输出向量 $Multi\text{-}Head(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})$:

$$head_i=Attention(\boldsymbol{Q}_i,\boldsymbol{K}_i,\boldsymbol{V}_i) \tag{5}$$
$$MultiHead(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=Concat(head_1,\cdots,head_h)\boldsymbol{W}^o,\ i=1,\cdots,h \tag{6}$$

其中,h 表示 Multi-Head Attention 的头数。

式(4)中计算两个输入向量之间的注意力得分为:

$$\begin{aligned}\alpha_{ij}&=\boldsymbol{Q}_i^{\mathrm{T}}\boldsymbol{K}_j\\ &=(\boldsymbol{W}_q m_i)^{\mathrm{T}}(\boldsymbol{W}_k m_j)\\ &=(\boldsymbol{W}_q(\boldsymbol{x}_i+\boldsymbol{u}_i))^{\mathrm{T}}(\boldsymbol{W}_k(\boldsymbol{x}_j+\boldsymbol{u}_j))\end{aligned} \tag{7}$$

其中,$\boldsymbol{W}_q$ 和$\boldsymbol{W}_k$ 分别是 Query 和 Key 的参数,$\boldsymbol{x}_i$ 和$\boldsymbol{x}_j$ 分别为第 i 个词和第 j 个词的词嵌入向量,$\boldsymbol{u}_i$和 $\boldsymbol{u}_j$ 分别为第 i 个位置和第 j 个位置的位置向量。

进一步将式(7)展开可得:

$$\alpha_{ij}=\underbrace{\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{W}_q^{\mathrm{T}}\boldsymbol{W}_k\boldsymbol{x}_j}_{(a)}+\underbrace{\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{W}_q^{\mathrm{T}}\boldsymbol{W}_k\boldsymbol{u}_j}_{(b)}+\underbrace{\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{W}_q^{\mathrm{T}}\boldsymbol{W}_k\boldsymbol{x}_j}_{(c)}+\underbrace{\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{W}_q^{\mathrm{T}}\boldsymbol{W}_k\boldsymbol{u}_j}_{(d)} \tag{8}$$

2.2 全局-局部注意网络的不足

文本是时序型数据,词与词之间的相对位置关系往往会影响整个句子的含义。通过分析可以发现,根据式(8)中的(d)项得到的位置向量只包含相对位置信息,而不包含方向信息,证明如下。

首先将式(2)展开为$\boldsymbol{u}_i$的矩阵表示形式:

$$\boldsymbol{u}_i=\begin{bmatrix}\sin(c_0 i)\\ \cos(c_0 i)\\ \sin(c_1 i)\\ \cos(c_1 i)\\ \cdots\sin(c_{\frac{d}{2}-1} i)\\ \cos(c_{\frac{d}{2}-1} i)\end{bmatrix} \tag{9}$$

其中,$c_k=\frac{1}{10000^{2k/d}}$是一个常量。

$$\begin{cases}\sin(\alpha+\beta)=\sin\alpha\cos\beta+\cos\alpha\sin\beta\\ \cos(\alpha+\beta)=\cos\alpha\cos\beta-\sin\alpha\sin\beta\end{cases} \tag{10}$$

然后根据上述三角函数的性质,可以得到第 i 个位置与第 j 个位置的位置向量的点积结果 $\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{u}_j$:

$$\begin{aligned}\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{u}_j&=\sum_{k=0}^{d/2-1}[\sin(c_k i)\sin(c_k j)+\cos(c_k i)\cos(c_k j)]\\ &=\sum_{k=0}^{d/2-1}\cos(c_k(i-j))\end{aligned} \tag{11}$$

由式(11)可知,$\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{u}_j$ 的结果只与相对位置 $i-j$ 有关,且会随着 $i-j$ 的递增而减小,从而起到表征相对位置的作用。但是,由于余弦函数的对称性,虽然其结果能够反映相对位置信息,但却无法区分方向,如式(12)所示:

$$\boldsymbol{u}_j^{\mathrm{T}}\boldsymbol{u}_i=\sum_{k=0}^{d/2-1}\cos(c_k(j-i))=\sum_{k=0}^{d/2-1}\cos(c_k(i-j))=\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{u}_j \tag{12}$$

由式(12)可知,当相对位置为 $j-i$ 时,$\boldsymbol{u}_j^{\mathrm{T}}\boldsymbol{u}_i=\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{u}_j$,无法获取方向信息。

进一步对式(8)的(d)项进行分析可知,虽然 $\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{u}_j$ 可以捕获相对位置信息,但实际上 $\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{W}_q^{\mathrm{T}}\boldsymbol{W}_k\boldsymbol{u}_j$ 是加入 $\boldsymbol{W}_q$ 和 $\boldsymbol{W}_k$ 矩阵后得到的,原本的距离感知能力被削弱,相对位置信息也不如原来明确。

基于上述分析,本文认为局部关系编码模块在对文本特征进行抽取时没有充分考虑到相对位置关系对谣言检测的影响,因此本文引入了一种新的相对位置编码方法对局部关系编码模块进行改进,以进一步提高谣言检测效果。

3 基于改进全局-局部注意力网络的谣言检测模型

基于 1.2 节的分析,本文提出了一种改进的全局-局部注意网络,下面分别对改进后的局部关系编码模块和网络架构中的全局关系编码模块进行说明。

3.1 基于改进相对位置编码的局部关系编码

首先,使用 Word2Vec 将数据集中原帖的文本转化为

词向量，将输入文本表示为 $\boldsymbol{M}=[\boldsymbol{m}_1,\cdots,\boldsymbol{m}_N]$，每条文本长度为 N。

在计算注意力得分时，本文根据 Dai 等[16]的方法，在式(8)的基础上作出如下 3 方面的改进，得到新的注意力权重计算公式，表示如下：

$$\alpha_{ij}=\underbrace{\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{W}_q^{\mathrm{T}}\boldsymbol{W}_{k,x}\boldsymbol{x}_j}_{(a)}+\underbrace{\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{W}_q^{\mathrm{T}}\boldsymbol{W}_{k,R}\boldsymbol{R}_{i-j}}_{(b)}+\underbrace{\boldsymbol{u}^{\mathrm{T}}\boldsymbol{W}_{k,x}\boldsymbol{x}_j}_{(c)}+\underbrace{\boldsymbol{v}^{\mathrm{T}}\boldsymbol{W}_{k,R}\boldsymbol{R}_{i-j}}_{(d)} \tag{13}$$

式(13)中的 3 点改进具体如下：

(1)用相对位置编码向量 $\boldsymbol{R}_{i-j}$ 替换 $\boldsymbol{u}_j$。

根据式(9)可以得出 $\boldsymbol{R}_{i-j}$ 的矩阵展开形式为：

$$\boldsymbol{R}_{i-j}=\begin{bmatrix}\sin(c_0(i-j))\\ \cos(c_0(i-j))\\ \cdots\\ \sin(c_{\frac{d}{2}-1}(i-j))\\ \cos(c_{\frac{d}{2}-1}(i-j))\end{bmatrix}=-\begin{bmatrix}-\sin(c_0(i-j))\\ \cos(c_0(i-j))\\ \cdots\\ -\sin(c_{\frac{d}{2}-1}(i-j))\\ \cos(c_{\frac{d}{2}-1}(i-j))\end{bmatrix}=-\boldsymbol{R}_{j-i} \tag{14}$$

式(12)中已知位置向量的点积结果 $\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{u}_j$ 存在的主要问题是无法区分方向信息。现从式(14)中可以看出，当相对位置为 $j-i$ 时，$\boldsymbol{R}_{j-i}=-\boldsymbol{R}_{i-j}$，可以区分方向信息。因此将式(8)的(b)项和(d)项中的 Key 向量的绝对位置编码 $\boldsymbol{u}_j$ 替换为相对于 Query 的相对位置编码 $\boldsymbol{R}_{i-j}$。

(2)用 $\boldsymbol{u}$ 替换式(8)中(c)项的 $\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{W}_q^{\mathrm{T}}$，用 $\boldsymbol{v}$ 替换式(8)中(d)项的 $\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{W}_q^{\mathrm{T}}$。

在考虑相对位置的情况下，不需要查询绝对位置，因此将式(8)的(c)项和(d)项中与 Query 相关的绝对位置向量 $\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{w}_q^{\mathrm{T}}$ 分别替换为与位置 i 无关的参数向量 $\boldsymbol{u}$ 和 $\boldsymbol{v}$，并且由于 $\boldsymbol{W}_q$ 是一个可训练的参数，因此 $\boldsymbol{u}$ 和 $\boldsymbol{v}$ 也是可训练的。

(3)分别用词嵌入映射矩阵 $\boldsymbol{W}_{k,x}$ 和位置嵌入映射矩阵 $\boldsymbol{W}_{k,R}$ 替换各项中的 $\boldsymbol{W}_k$。

在对 Query 和 Key 分别进行线性映射时，Query 对应 $\boldsymbol{W}_q$ 矩阵，Key 对应 $\boldsymbol{W}_k$ 矩阵，将式(7)中括号里的内容进一步展开，可以得到单词嵌入和单词的位置嵌入与 $\boldsymbol{W}_q$ 和 $\boldsymbol{W}_k$ 的关系，如式(15)所示：

$$\begin{aligned}\alpha_{ij}&=(\boldsymbol{W}_q(\boldsymbol{x}_i+\boldsymbol{u}_i))^{\mathrm{T}}(\boldsymbol{W}_k(\boldsymbol{x}_j+\boldsymbol{u}_j))\\&=(\boldsymbol{W}_q\boldsymbol{x}_i+\boldsymbol{W}_q\boldsymbol{u}_i)^{\mathrm{T}}(\boldsymbol{W}_k\boldsymbol{x}_j+\boldsymbol{W}_k\boldsymbol{u}_j)\end{aligned} \tag{15}$$

从式(15)中可以看出，Query 和 Key 的单词嵌入和单词的位置嵌入都是采用同样的线性变换得到。引入相对位置编码后，为了更好地区分两者，将 $\boldsymbol{W}_k$ 拆分成两个矩阵，其中 $\boldsymbol{W}_{k,x}$ 对应 Key 的词嵌入线性映射矩阵，$\boldsymbol{W}_{k,R}$ 对应位置嵌入的线性映射矩阵。

然后，将改进后的注意力得分用于计算 Self-Attention，再利用 Multi-Head Attention 对其结果进行拼接，作为输出向量。

最后，将注意力机制的输出向量经 CNN 的卷积层和池化层进行优化，将优化后的结果作为局部特征向量 $\tilde{\boldsymbol{m}}$。

3.2 全局特征向量提取及谣言检测

3.1 节对局部关系编码的文本特征进行了抽取，本节则主要对发布和转发帖子的用户构成的图结构进行分析，从而得到全局特征向量。

基于帖子的原始发布用户和转发用户之间的关系构造图 $G=(V,E)$，其中 v 是节点集，即发布和转发帖子的用户构成的集合，节点的数量为发布和转发用户数量的总和；E 是节点间的边集，即发布用户和转发用户之间的边构成的集合。

本文利用图注意力网络[15]学习图中的结构信息。具体流程如下。

首先，输入每个用户节点的特征向量集 $\boldsymbol{h}$，该特征向量集由每个用户所占的权重组成，权重是在用户转发帖子的时刻和原帖发布时刻的差值基础上计算得到。

$$\boldsymbol{h}=\{h_1,h_2,\cdots,h_N\},h_i\in R^F \tag{16}$$

其中，N 表示节点个数，F 表示特征向量维度。

然后，使用 Self-Attention 计算每个节点的邻居节点对当前节点的贡献度，归一化后的注意力系数为：

$$\alpha_{ij}=\frac{\exp(\mathrm{LeakyRelu}(\boldsymbol{\alpha}^{\mathrm{T}}(\boldsymbol{W}\boldsymbol{h}_i\parallel\boldsymbol{W}\boldsymbol{h}_j)))}{\sum_{k\in N_i}\exp(\mathrm{LeakyRelu}(\boldsymbol{\alpha}^{\mathrm{T}}(\boldsymbol{W}\boldsymbol{h}_i\parallel\boldsymbol{W}\boldsymbol{h}_k)))} \tag{17}$$

其中，N_i 为邻居节点集，Leaky Relu 为激活函数。

将归一化后的注意力系数与其对应的特征进行线性组合，作为每个节点的最终输出特征，并将所得结果采用 Multi-Head Attention 进行拼接，输出特征向量为：

$$\boldsymbol{h}_i'=\Big\Vert_{k=1}^{K}\sigma\Big(\sum_{j\in N_i}\alpha_{ij}^k\boldsymbol{W}^k\boldsymbol{h}_j\Big) \tag{18}$$

其中，k 表示 Multi-Head Attention 的头数，$\boldsymbol{W}$ 表示权重矩阵，$j\in N_i$ 中的 j 表示所有与 i 相邻的节点。

根据式(18)，将输出的全局特征向量表示为 $\boldsymbol{h}'$：

$$\boldsymbol{h}'=\{h_1',h_2',\cdots,h_N'\},h_i'\in R^{F'} \tag{19}$$

其中，F' 表示新的节点特征向量维度。

最后，将局部特征向量 $\tilde{\boldsymbol{m}}$ 和全局特征向量 $\boldsymbol{h}'$ 输入 softmax 层进行分类：

$$p_i=\mathrm{softmax}(\boldsymbol{W}(\tilde{\boldsymbol{m}};\boldsymbol{h}')+b) \tag{20}$$

其中，$\boldsymbol{W}$ 表示权重矩阵，b 为偏置项。

4 实验与结果分析

4.1 数据集

本文采用 Ma 等[9]于 2016 年公开的新浪微博数据集，该数据集总共包括 4 664 个帖子以及相应的标签。表 1 列出了该数据集的统计数据。

表 1 数据集的统计数据

Table 1 Statisticsof datasets

数据集	微博
events	4664
rumor	2313
non-rumor	2351
Microblogs	3805656
Users	2746818

4.2 评价指标与参数设置

为了评估本文模型的有效性，本文从准确率(Accuracy)、精确率(Precision)、召回率(Recall)和 F1 值(F1-score)4 个方面进行评估，分类结果的混淆矩阵如表 2 所列。

表 2 分类结果的混淆矩阵

Table 2 Confusion matrix of classification results

数据集	实际为谣言	实际为非谣言
被分类为谣言	TN	FN
被分类为非谣言	FP	TP

真正例(True Positive,TP)表示将正样本正确预测为正样本;假正例(False Positive,FP)表示将负样本错误预测为正样本;假负例(False Negative,FN)表示将正样本错误预测为负样本;真负例(True Negative,TN)表示将负样本正确预测为负样本。

(1)准确率:所有预测正确的结果占所有结果的比重。

$$Accuracy=\frac{TP+TN}{TP+TN+FP+FN} \tag{21}$$

(2)精确率:正确预测为正类的结果占所有预测为正类结果的比重。

$$Precision=\frac{TP}{TP+FP} \tag{22}$$

(3)召回率:正确预测为正类的结果占所有实际为正类的结果的比重。

$$Recall=\frac{TP}{TP+FN} \tag{23}$$

(4)F1 值:精确率和召回率的调和均值。

$$\frac{2}{F_1}=\frac{1}{Precision}+\frac{1}{Recall} \tag{24}$$

本文所有实验代码均在 PyTorch 框架下实现,超参数设置如表 3 所列。

表 3 超参数设置

Table 3 Setting of super parameters

超参数	数值
learning rate	1×10^{-3}
注意力头数	8
batch size	64
dropout	0.5
卷积核大小	(3,4,5)

4.3 对比实验与结果分析

为了评估本文方法的效果,在 Ma 等[12]提出的数据集上,将本文方法与其他主流的谣言检测模型进行对比实验。用于对比的模型如下。

(1)DTC 模型[4]:通过提取文本情感分数、包含 URL 的微博数、用户注册天数、发布的微博数等特征,采用决策树分类器进行谣言检测。

(2)SVM-RBF 模型[5]:在文本、用户和传播特征的基础上提出客户端类型和事件发生的地理位置两个新特征,并利用 SVM 分类器进行分类。

(3)SVM-TS 模型[17]:提出基于内核的传播树方法,通过评估传播树之间的相似性来识别谣言。

(4)GRU 模型[9]:通过 GRU 网络学习微博事件连续表示来识别谣言。

(5)PPC 模型[18]:基于用户的谣言传播路径,结合 RNN 和 CNN 进行谣言检测。

(6)GLAN 模型[13]:将基于文本特征的局部语义信息和用户特征的全局结构信息结合起来编码,用于谣言检测。

实验结果如表 4 所列。

表 4 实验结果

Table 4 Experimental results

模型	类别	准确率	精确率	召回率	F1
DTC	R	0.831	0.847	0.815	0.831
	N		0.815	0.847	0.830
SVM-RBF	R	0.818	0.822	0.812	0.817
	N		0.815	0.824	0.819
SVM-TS	R	0.857	0.839	0.885	0.861
	N		0.878	0.830	0.857
GRU	R	0.910	0.876	0.956	0.914
	N		0.952	0.864	0.906
PPC	R	0.921	0.896	0.962	0.923
	N		0.949	0.889	0.918
GLAN	R	0.946	0.943	0.948	0.945
	N		0.949	0.943	0.946
Our Method	R	0.950	0.947	0.952	0.949
	N		0.952	0.947	0.950

注:R 为谣言;N 为非谣言

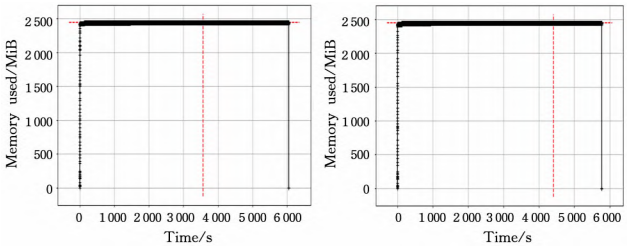
从表 4 的实验结果可以看出,在相同数据集上,DTC,SVM-RBF,SVM-TS 等基于传统机器学习的方法分类效果较差,原因是人工构造特征工程量较大,影响因素较多;而 GRU,PPC,GLAN 等基于深度学习的方法分类效果较好,这是因为深度神经网络能够自动学习模型更深层次的特征。GLAN 在深度学习模型的基础上,将文本的局部语义信息和全局结构信息很好地融合在一起,使得谣言检测的准确率得到了显著提高。

考虑到文本中词与词之间的位置关系对谣言检测的影响,本文在 GLAN 的基础上引入相对位置编码,使其能够更准确地提取谣言中文本的语义信息和位置信息并聚合,并将该局部特征与描述转发行为的全局特征相结合,进而提升对谣言的检测效果,使模型准确率从 0.946 提高到 0.950,精确率、召回率和 F1 值较 GLAN 模型均有提升,验证了本文模型的有效性。

4.4 模型性能分析

除上述评价指标外,为了对模型进行更全面的分析,实验利用 memory_profiler 模块,从时间维度对比分析了本文模型与 GLAN 模型在训练和测试阶段的内存使用情况,对比结果如图 3 和图 4 所示。

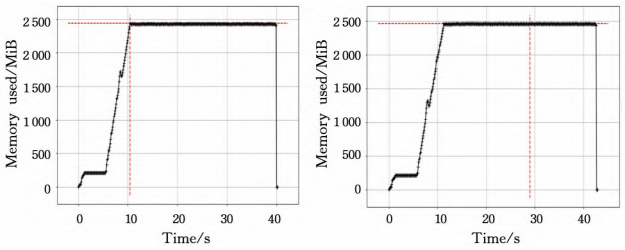
如图 3 和图 4 所示,横轴表示运行时间,纵轴表示内存占用情况。从图 3(a)和图 3(b)中可以看出,在训练阶段,虽然本文模型比 GLAN 模型的运行时间短,但内存占用更多;从图 4(a)和图 4(b)中可以看出,在测试阶段,本文模型与 GLAN 模型的运行时间相差较小,本文模型的内存占用为 2450.1 MiB,GLAN 模型的内存占用为 2425.8 MiB。实验结果表明本文模型的内存占用量相对增多,后续有待进一步研究改进。



(a)GLAN 模型 (b)本文模型

图 3 训练阶段模型内存占用情况

Fig. 3 Memory usage of model in training phase



(a)GLAN 模型 (b)本文模型

图 4 测试阶段模型内存占用情况

Fig. 4 Memory usage of model in test phase

结束语 为了提高谣言检测的精度,考虑到文本中词与词之间的位置关系对谣言检测的影响,本文提出了基于改进相对位置编码的谣言检测模型,该方法能够更准确地提取谣言中文本的语义信息和位置信息,更好地区分谣言与非谣言的文本特征,并与描述转发行为的全局结构特征相结合,最后通过分类器对微博数据集进行分类。

在未来的工作中,可以考虑将数据集中其他类型的用户信息加入图结构,使其可以利用更多的信息来区分谣言与非谣言,更好地提高谣言检测效果。

参 考 文 献

[1] LIU Z Y,ZHANG L,TU C C,et al. Statistical and Semantic Analysis of Rumors in Chinese Social Media[J]. Scientia Sinica Informationis,2015,45(12):1536-1546.

[2] LIU Z Y,SONG C H,YANG C. Early Detection of Rumors in Social Media[J]. Global Media Journal,2018,5(4):65-80.

[3] China Internet Network Information Center. The 47th statistical report on internet development in China[R]. Beijing:CNNIC,2020.

[4] CASTILLO C,MENDOZA M,POBLETE B. Information Credibility on Twitter[C]// Proceedings of the 20th International Conference on World Wide Web. ACM,2011:675-684.

[5] YANG F,LIU Y,YU X,et al. Automatic Detection of Rumor on Sina Weibo[C]// Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics. ACM,2012:13-20.

[6] KWON S,CHA M,JUNG K,et al. ProminentFeatures of Rumor Propagation in Online Social Media[C]// Proceedings of the 13th IEEE International Conference on Data Mining. ICDM,2013:1103-1108.

[7] WU K,YANG S,ZHU K Q. False Rumors Detection on Sina Weibo by Propagation Structures[C]// Proceedings of the 31st IEEE International Conferenceon Data Engineering. ICDE,2015:651-662.

[8] MA J,GAO W,WONG K F. Detect Rumors in Microblog Posts Using Propagation Structure via Kernel Leaming[C]// Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. ACL,2017:708-717.

[9] 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. IJCAI,2016:3818-3824.

[10] 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. Melbourne. ACL,2018:1980-1989.

[11] MA J,GAO W. Debunking Rumors on Twitter with Tree Transformer[C]// Proceedings of the 28th International Conference on Computational Linguistics. COLING,2020:5455-5466.

[12] 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. AAAI,2020:549-556.

[13] 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 19th IEEE International Conference on Data Mining. ICDM,2019:796-805.

[14] 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. NIPS,2017:5998-6008.

[15] VELICKOVIC P,GUILLEM C,CASANOVA A,et al. Graph Attention Networks[J]. arXiv:1710.10903,2017.

[16] DAI Z H,YANF Z L,YANG Y M,et al. Transformer-XL:Attentive Language Models Beyond a Fixed-length Context[C]// Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. ACL,2019:2978-2989.

[17] 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. ACM,2015:1751-1754.

[18] 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 32nd AAAI Conference on Artificial Intelligence. AAAI,2018:354-361.

JIANG Meng-han, born in 1996, postgraduate. Her main research interests include natural language processing and so on.

LI Shao-mei, born in 1982, Ph.D, associate professor. Her main research interests include natural language processing and so on.

(责任编辑:何杨)