

时序演化特征挖掘的特定域社会事件检测方法

赵东旭,张鑫*,潘岩,段嘉顺
(国防科技大学 系统工程学院,湖南 长沙 410073)

摘要:针对隐式关联关系挖掘不充分、未考虑时序演化因素等问题,提出时序演化特征挖掘的特定域社会事件检测方法。将数据按时间划分切片,通过考虑不同来源的重复事件记录对检测的影响,构造实体交互图,降低数据库误差影响;改进多关系图卷积网络,依据交互关系更新历史演化序列的图结构信息,通过注意力机制关注重要特征,得到序列单元全局表示,实现隐式关联的充分挖掘;基于循环神经网络提取时序演化特征,得到序列全局表示,实现时序演化因素的挖掘。实验结果表明,该方法能够适用于特定域社会事件检测任务,较现有方法效果更优。

关键词:特征挖掘;图卷积网络;注意力机制;时序演化特征;事件检测

中图分类号:TP183 **文献标志码:**A **文章编号:**1001-2486(2025)05-186-11



Detecting domain-specific social events via temporal evolution feature mining

ZHAO Dongxu, ZHANG Xin*, PAN Yan, DUAN Jiashun

(College of Systems Engineering, National University of Defense Technology, Changsha 410073, China)

Abstract: In view of the insufficient mining of implicit associative relationships and the problem of neglecting the temporal evolution factor, a domain-specific social event detection method via temporal evolution feature mining was proposed. The data was sliced by time and an entity interaction graph was constructed by considering the impact of duplicate event records from different sources on detection, in order to reduce the influence of database errors. Multi-relational graph convolutional network was improved, and the graph structure information of historical evolution sequence was updated by interaction relationships. Attention mechanism was used to learn core features to obtain global embedding of sequence units. Implicit association was mined sufficiently. Based on recurrent neural network, temporal evolution features were extracted to obtain the global embedding and the temporal evolution factor was mined effectively. Experiment results show that the proposed method can be applied to domain-specific social event detection task, which is better than existing methods.

Keywords: feature mining; graph convolutional networks; attention mechanism; temporal evolution features; event detection

社会事件是指在某个时间、地点聚集的群体性活动^[1]。大数据时代,从复杂多元的互联网获取信息已经成为人们了解社会的重要手段,但信息质量良莠不齐,如何从中检测感兴趣的社会事件,对跟踪事件发展、处置突发情况具有十分重要的意义。相关任务按是否针对特定类型事件可分为开放域(open domain)与特定域(domain-specific)事件检测^[2-4]。开放域事件检测不针对特定事件,通常利用时间或信号突变,将趋势事件

与非趋势事件区分开来。特定域事件检测针对感兴趣的特定事件(如抗议示威、地震等),利用机器学习、数据挖掘等技术实现,挖掘的是感兴趣事件的发生机理,检测的结果不涉及其他类型事件。因此,特定域社会事件检测的实际应用价值更大^[5]。考虑到群体抗议事件往往具有突发性,且发生后冲击社会稳定、影响民众生活,本论文以该类事件为例,研究特定域社会事件检测。

相关研究按所用数据形式和来源不同可分

收稿日期:2023-07-25

基金项目:国家自然科学基金资助项目(62102431);国防科技大学自主科研计划基金资助项目(ZK21-32);信息系统工程重点实验室基金资助项目(6142101220209)

第一作者:赵东旭(1996—),男,黑龙江齐齐哈尔人,硕士研究生,E-mail:dxzhao_nudt@foxmail.com

*通信作者:张鑫(1978—),男,四川南江人,研究员,博士,硕士生导师,E-mail:shinezhang_nudt@qq.com

引用格式:赵东旭,张鑫,潘岩,等. 时序演化特征挖掘的特定域社会事件检测方法[J]. 国防科技大学学报, 2025, 47(5): 186-196.

Citation: ZHAO D X, ZHANG X, PAN Y, et al. Detecting domain-specific social events via temporal evolution feature mining[J]. Journal of National University of Defense Technology, 2025, 47(5): 186-196.

两类:

一是面向非结构化数据的检测方法,其主要利用新闻文本或社交帖文作为输入数据。例如, Sakaki 等^[6]提出一种基于支持向量机(support vector machine, SVM)的方法,识别有关日本地震的推文,并通过实验证实了所提方法的有效性。Lee 等^[7]构建半监督卷积神经网络对推文中提到的不良药物事件进行检测。蒋沂桔^[8]通过多头注意力机制的图卷积网络与门控机制将语义与依存关系嵌入融合,实现多类型事件检测。Afyouni 等^[1]设计了混合学习模型 Deep-Eware,通过引入用于事件发现的纯增量方法,集成 KeyBERT 语义关键字生成工具,在 Twitter 上验证了在线检测性能。Carta 等^[9]提出了面向金融的事件检测方法,通过专用词典实现词嵌入,设计分层聚类算法实现检测。这类方法通常基于文本嵌入或其他特征表示,局限主要包括:①容易受到文本中包含的非感兴趣事件信息干扰;②未考虑利用感兴趣事件发生前各类征候事件及其演化过程;③检测结果中通常缺少事发时间、地点、参与者等关键要素,还需进行命名实体识别、时间解析、地理位置估计等复杂后处理,才能推断出这些信息。

二是面向结构化数据的检测方法,主要利用 GDELT^[10](global data on event, location and tone)、ICEWS^[11](integrated conflict early warning system)等事件库中结构化记录识别感兴趣的事件。如 GDELT 可通过直接检索(retrieving databases directly, RED)实现事件检测。Keneshloo 等^[12]首次基于 GDELT 使用频繁子图方法对巴西等国的政治危机进行检测。Qiao 等^[13]完善频繁子图挖掘算法(detection method based on frequent subgraphs, FSM),利用频繁子图构造感兴趣事件的特征选择集,以此进行特征表示,通过 SVM 实现抗议事件检测。Azeemi 等^[14]分别通过对地理位置与标题编码嵌入,实现了多类型感兴趣事件的检测,本质是基于短文本嵌入。不难看出,虽然面向结构化数据方法研究较少,但优势明显:①可用特征更丰富,结构化数据通常提供参与者、时间、地点、主题、其他附件等信息,能为检测提供更多数据支撑;②以字段为桥梁容易将结构化记录关联汇聚起来,便于挖掘感兴趣事件的关联与演化过程。然而不足也较为明显:①隐式关联挖掘不充分。隐式关联是指未直接决定事件发生的参与者间交互演化关系,如在融入了周边节点与边信息的关键子图中,每个节点融入的周边交互关系均间接决定了事件的最终发生。而显式关联则

是直接决定事件发生的核心交互演化模式,其并没有融入周边交互信息,如上述频繁子图。由于现有研究挖掘的是频繁子图,且没有利用特征挖掘与表示能力更强的深度学习技术,因此无法充分挖掘隐式关联。②未考虑时序演化因素。由于事件通常基于酝酿演化而最终发生,因此其随时间的变化特征也需纳入考虑。

随着深度学习的迅猛发展,图注意力网络^[15]、多关系图卷积网络^[16-17]等图神经网络通过聚合周边节点与边信息实现特征更新,在挖掘要素依赖关系上具有显著优势。对于多关系交互演化模式,基于组合的多关系图卷积网络^[17](composition-based multi-relational graph convolutional networks, CompGCN)在实验中表现出色,因此能够将该网络接入分类器实现检测。

基于上述调研分析,本论文以群体抗议事件为例,提出时序演化特征挖掘的特定域社会事件检测方法(detecting domain-specific social events via temporal evolution feature mining, TEFM),利用图神经网络来建模和挖掘演化特征,实现高精度检测。具体地,选用当前开放程度最高、数据时效性最强、学术影响力最大的 GDELT 作为数据来源;设计了交互关系挖掘与基于注意力机制的图表示算法,解决了隐式关联关系挖掘不充分的问题;设计了时序演化挖掘算法,考虑了时序演化因素对检测的影响;在构建的数据集上设计系列实验,以验证检测方法的有效性。

1 问题建模

考虑到特定域社会事件检测基于社会事件是经酝酿演化而最终发生的假设,研究目标是通过对感兴趣事件发生前参与群体的交互行为进行挖掘,判断当下时刻感兴趣事件发生的概率,本文将该问题建模为一个有监督的二分类问题。这样建模的优势在于能够充分利用事件发生前参与群体的交互演化关系推断事件发生概率,相比于事件记录层面的检索、分类(如 SVM)或聚类更符合逻辑,这是由于结构化数据存在以下不足:①对其他无关新闻事件或者历史事件的大量提及;②数据库在机器翻译、事件抽取、事件编码中引入的误差;③非感兴趣事件的类别错划,如抗议类事件中,国家间的声张(declaration)等也常被归类为抗议事件。因此,对该问题的建模方式概括如下:给定目标地域在当下时刻 T 的历史事件集合 $E_{(T-l+1,T)}$,通过将其按天划分时间切片得到 $G_{T-l+1}, G_{T-l+2}, \dots, G_T$,构建时间跨度为 l 的历史

演化序列 $X_T = \{G_{T-l+1}, G_{T-l+2}, \dots, G_T\}$ 作为模型输入,模型的输出为目标地域在当下 T 时刻发生群体性抗议事件的概率 $P(Y_T|X_T)$ 。总结起来,本文关注的是,利用特定地域(如某个城市)在特定时间 T 之前的历史演化序列作为输入,判断该地在时间 T 发生感兴趣社会事件(如群体性抗议事件)的可能性,形式化定义为:

$$P(Y_T|X_T) = f(X_T) \quad (1)$$

对上述定义给出实例化阐述:实体交互图 G_T 是由目标地域在时间戳 T 日的事件元组 ($\text{Actor1 Name} \xrightarrow{\text{EventCode}} \text{Actor2 Name}$) 构建的多关系有向图 $G_T = (V(S, O), R, T)$, 其中 S 表示全部事件元组交互行为的发起方(施动者)集合, O 表示全部事件元组交互行为的接受方(受动者)集合, V 表示由 S 与 O 组成的集合, R 表示对应于 S 与 O 的全部事件元组交互行为集合。由于 S 与 O 之间可能存在多个交互,因此 G_T 是异构图。图 1 展示了实体交互图示例,不同颜色代表不同节点,出于美观考虑,图中对边类型未加区分。

历史演化序列 X_T 是由时间戳 T 前跨度为 l 的全部实体交互图按时间先后组成的序列,蕴含感兴趣事件在时间跨度 l 内的隐式演化特征。图 2 展示了感兴趣事件在时间戳 T 的历史演化序列。由于构成历史演化序列各单元数据的不同,因此序列上各实体交互图的结构也不一致。

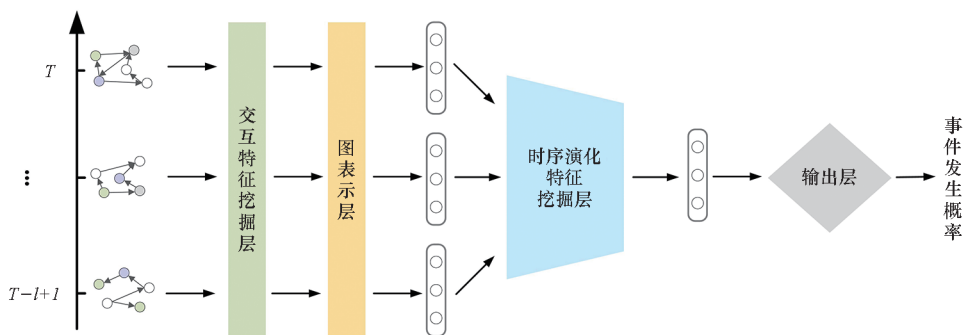


图 3 模型总体框架

Fig. 3 Overall model framework

2.1 交互特征挖掘层

第一步,将数据集的全部事件记录以天为时间切片划分实验数据,以参与者信息为桥梁、交互关系为连接,构建当日实体交互图 G_T 。第二步,依据时间跨度 l 向前获取相应个数的实体交互图,并按照时间顺序组建历史演化序列 X_T 。第三步,使用 Xavier 初始化方法^[18]为数据集中所有的参与者(节点)与事件类型(边)分别生成初始化嵌入 $H_V \in \mathbb{R}^{n_1 \times d}$ 与 $H_R \in \mathbb{R}^{n_2 \times d}$, 其中 n_1 与 n_2 分别

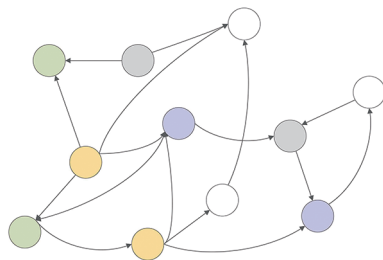


图 1 实体交互图示例(局部)

Fig. 1 Example of entity interaction graph (part)

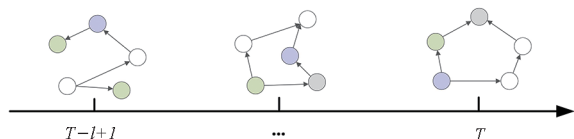


图 2 历史演化序列

Fig. 2 Historical evolution sequence

2 算法模型与分析

TEFM 主要包括交互特征挖掘层、图表示层、时序演化特征挖掘层与输出层,总体架构如图 3 所示,步骤包括:将历史演化序列输入交互特征挖掘层,更新序列单元的节点与边信息;将更新后的序列单元通过图表示层,得到其全局表示;将上述全局表示按照时间顺序堆叠为序列嵌入再输入时序演化特征挖掘层,获取序列的全局表示;将上述序列的全局表示映射为事件发生的概率。

为节点与边在数据集中的种类数量, d 为初始化嵌入的特征维度,初始化过程严格记录种类索引。第四步,模块层对序列中每个实体交互图的结构信息进行特征更新,充分捕获邻居节点与边的信息,挖掘交互关系。这一部分的主体结构为 CompGCN,由于构建的实体交互图是单向传播的,因此有必要去除对反向与自循环边的更新,以符合任务实际,减少参数使用。图 4 展示了时间戳 T 日实体交互图 G_T 的局部更新计算过程,黑色实线为图中蓝色节点的聚合更新过程,灰色实线

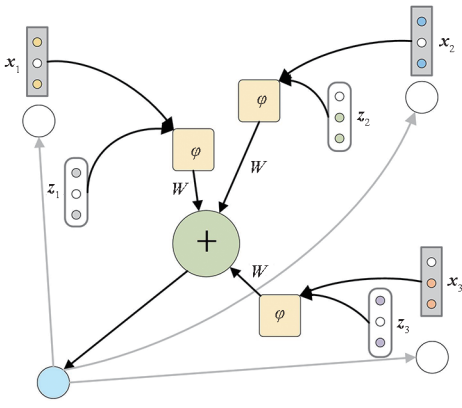


图4 交互特征挖掘过程

Fig. 4 Process of interaction feature mining

为图中各节点的交互关系, x_1, x_2, x_3 为周边节点的特征, z_1, z_2, z_3 为周边邻接关系的特征。定义 $v \in N_{\text{index}}$ 是 G_T 中的蓝色节点, N_{index} 为模块层输入 G_T 的全部头尾节点索引组合集合, $r_{\text{all}} \in R_{\text{index}}$ 是 v 的所有邻接关系, R_{index} 为模块层输入 G_T 中全部边索引集合, $N(v)$ 是节点 v 在关系 r_{all} 下的邻接节点集合, h_v 是节点 v 的表示, 则节点 v 在聚合邻接节点与边的过程如下:

$$h_v = f\left(\sum_{(u,r) \in N(v)} W h_u\right) \quad (2)$$

式中, u, r 分别为 v 的邻居集合中的节点与相应的边, W 是投影矩阵。 h_u 是综合考虑了节点 u 和关系 r 的影响而通过某种组合方式结合在一起的向量表示, 具体的组合方式如下:

$$h_u = \varphi(x_u, z_r) \quad (3)$$

式中, $\varphi(x_u, z_r)$ 表示基于循环相关性的组合方式, x_u 是 u 的初始特征表示, z_r 是 r 的初始特征表示。节点特征更新后, 边也同步更新, 计算过程如下:

$$h_r = W_{\text{rel}} z_r \quad (4)$$

式中, W_{rel} 是将边映射到节点空间的映射变换矩阵, 便于对节点与边统一计算。更新后的节点嵌入与边嵌入分别表示为 $H_V^1 \in \mathbb{R}^{n_1 \times d}$ 与 $H_R^1 \in \mathbb{R}^{n_2 \times d}$ 。

至此, 模块层对交互关系的挖掘学习过程全部结束, 更新后的节点与边均隐含着邻居节点与边的知识信息(隐式关联关系), 在一定程度上能够表征参与群体的交互行为, 是形成实体交互图全局表示的基础。

2.2 图表示层

在更新节点嵌入与边嵌入之后, 需要将 G_T 进行融合得到其全局表示。而 G_T 内的事件元组存在一定的重复记录, 其主要来自两个方面, 即数据库自身以及不同的新闻源。前者多是由于数据库

在采集过程中因网页内容提取算法、机器翻译等带来的误差, 通常在数据获取过程中筛选去除; 后者虽然事件元组相同, 但新闻来源不同, 是事件重要性的侧面反映, 因此需要将其考虑其中。

图5展示了图表示层的工作原理。首先, 模块层获取输入交互特征挖掘层的头、尾节点索引与边索引, 分别构建索引张量 i_s, i_o 与 i_r 。其次, 依据节点与边的索引张量分别从更新后的节点嵌入 H_V^1 与边嵌入 H_R^1 获取对应的特征表示, 同时将其相同索引的特征表示相加, 得到 S^0, O^0 与 R^0 。再次, 对照 H_V^1 与 H_R^1 的维度和位置对 S^0, O^0 与 R^0 进行放缩, 将未在 G_T 中出现的节点与边进行遮挡(即置为0特征), 出现的则按照位置对应覆盖, 得到 $S^1 \in \mathbb{R}^{n_1 \times d}, O^1 \in \mathbb{R}^{n_1 \times d}$ 与 $R^1 \in \mathbb{R}^{n_2 \times d}$ 。最后, 引入自注意力机制^[19]分别学习头、尾节点与边的自身间注意力关系, 以便更好突出重要事件元组的作用。具体计算过程如下:

$$\begin{cases} Q = A^T W_q \\ K = A^T W_k \\ V = A^T W_v \end{cases} \quad (5)$$

式中, $W_q, W_k, W_v \in \mathbb{R}^{i \in \{n_1, n_1, n_2\} \times d_1}$ 表示3个不同的权重矩阵, A 为自注意力网络的输入, d_1 为矩阵维度, $Q, K, V \in \mathbb{R}^{d \times d_1}$ 分别是由输入得到的查询矩阵、键矩阵与值矩阵。由此得到经自注意力网络更新后的输出 $A' \in \mathbb{R}^{d \times d_1}$:

$$A' = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (6)$$

式中, $\sqrt{d_k}$ 为放缩因子, 避免在反向传播中梯度过小。公式中除去 V 的部分表示自注意力权重矩阵, 对其按行求模并归一化得到一维权重分数 $t \in \mathbb{R}^{1 \times d}$, 进而得到 A' 的一维全局表示向量:

$$p_{i \in \{s, o, r\}} = (tA') \quad (7)$$

式中, $p_i \in \mathbb{R}^{d_1}$ 为头、尾节点与边经自注意力网络的最终输出。当式(5)中 A 为 S^1 时, p_i 为头节点输出 p_s ; 当 A 为 O^1, R^1 的情况以此类推。模块层的最终输出为上述输出的拼接向量, 即时间戳 T

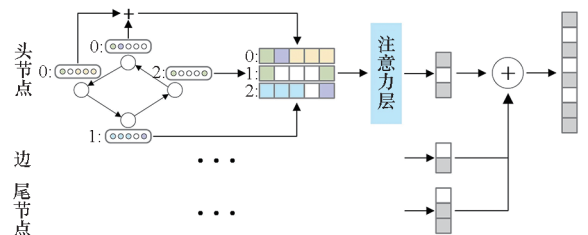


图5 图全局表示计算过程

Fig. 5 Process of graph global embedding

日实体交互图 G_T 的最终全局表示向量,记为 $\boldsymbol{p}$, 计算公式如下:

$$\boldsymbol{p} = \text{concat}([\boldsymbol{p}_s, \boldsymbol{p}_o, \boldsymbol{p}_r]) \in \mathbb{R}^{3 \times d_1} \quad (8)$$

经图表示层的计算后,上述过程为序列单元全局表示中的各节点与边均计算出了不同的贡献值,突出了重要头、尾节点与边对全局表示的贡献度,能够实现显式关联关系的精细挖掘。

2.3 时序演化特征挖掘层

模块层输入是由历史演化序列 X_T 中每个实体交互图的全局表示按时间切片先后顺序堆叠构成,记为 $\boldsymbol{X}_T$ 。模块层的核心是由一层双向长短期记忆网络 (bidirectional long short-term memory, Bi_LSTM) 构成的循环神经网络^[20],在序列建模上具有一定优势,兼具长短时记忆功能,避免在训练过程出现梯度消失或爆炸问题,工作示意如图 6 所示。

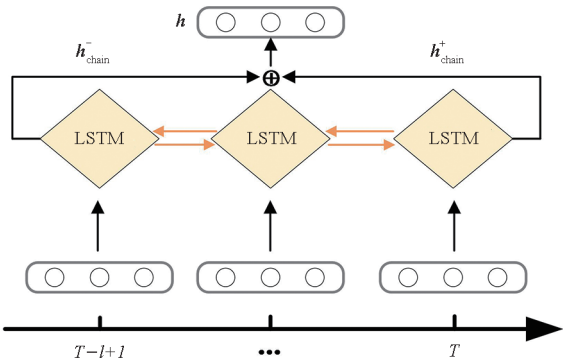


图 6 时序演化特征挖掘层

Fig. 6 Temporal feature evolution mining layer

模块层首先将更新后的历史演化序列 $\boldsymbol{X}_T$ 输入循环神经网络,经过序列学习得到网络的双向输出 $\boldsymbol{h}_{\text{chain}}^+$ 与 $\boldsymbol{h}_{\text{chain}}^-$ 。而后将上述两个输出拼接得到 $\boldsymbol{h}$ 作为时序演化特征挖掘层的最终输出。

2.4 输出层

将上述输出接入全连接网络,经 sigmoid 函数映射为事件发生概率,至此完成模型前向传播。对于模型优化,通过比较检测概率值与真实标签的差距实施评估优化,具体是采用基于 L2 正则化的二元交叉熵(binary cross entropy, BCE)。

3 实验与分析

3.1 数据集构建

考虑到数据的权威性,选取维基百科中明确记录且在全球主流媒体曾广泛报道、关注度高的事件词条,并适当扩展时间范围,通过限定地理位置、时间等检索条件从 GDELT 中获取数据至本

地,具体如算法 1 所示。

算法 1 数据获取算法

Alg. 1 Data acquisition algorithm

输入:GDELT	
输出:存储至本地的事件数据记录	
1. 与 GDELT 建立连接 2. while start_time <= end_time do 3. 通过限定地点、时间(按天)等设置检索条件 4. 获取记录(Actor1Name, EventCode, Actor2Name) 5. 去除参与者均为空值的记录 6. 去除新闻来源一致的重复事件记录 7. 将每日的全部事件记录存储本地 8. start_time = start_time + 1 9. end	

得到的数据集包括:2020 年发生在美国明尼苏达州的“抗议乔治·弗洛伊德之死”(US);2020 至 2021 年发生在印度新德里的“农民抗议”(IN);2022 年围绕俄乌冲突发生在基辅的抗议(UKR)。其中,US 因抗议主要发生在 5 至 7 月,且遍布美国多地,故补充了俄亥俄州、加利福尼亚州以及佐治亚州的数据。数据集时间跨度情况如表 1 所示。

表 1 数据集时间跨度情况
Tab. 1 Time interval of datasets

数据集	地理位置	时间跨度
US	明尼苏达	2020 年 1 月 2 日—2020 年 7 月 31 日
	俄亥俄	2020 年 5 月 26 日—2020 年 7 月 31 日
	加利福尼亚	2020 年 5 月 26 日—2020 年 7 月 31 日
	佐治亚	2020 年 5 月 26 日—2020 年 7 月 31 日
IN	新德里	2020 年 9 月 1 日—2021 年 8 月 30 日
UKR	基辅	2022 年 1 月 1 日—2022 年 12 月 31 日

在正负例定义方面,根据每日抗议事件的发生情况作为正负例区分的依据。具体地,统计事件类型编码为抗议类的事件记录频率阈值,大于阈值的当日记正例,反之记负例。阈值的设定参考在维基百科中有明确记录的感兴趣事件,统计事件发生当日事件类型编码为 14 的事件记录出现频率,由频率的均值最终确定。具体如图 7 所示,图中展示了 US 实验数据集的正负例划分情况,图中的蓝色虚线表示设定的阈值。

表 2 展示了数据集在划分正负例后的详细情况,数据按天存储,不含数据集时间跨度之外用于构造序列的数据。因此,在构建历史演化序列时

还需依据序列长度向前增加相应的数据。

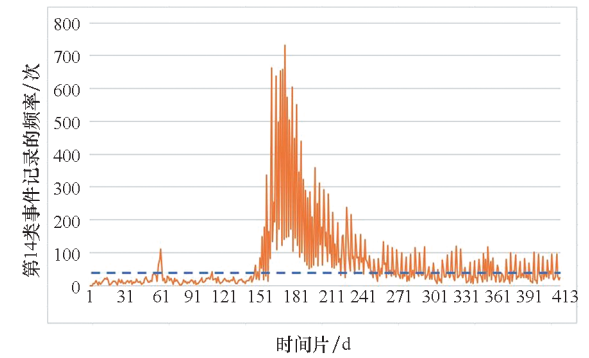


图7 正负例划分示例

Fig.7 Division of positive and negative examples

表2 数据集详细情况

Tab.2 Details of datasets

数据集	正例数	负例数	时间片/d	记录数
US	152	261	413	581 170
IN	168	196	364	325 070
UKR	160	205	365	455 602

3.2 评价指标、对比方法及实验超参数

1)评价指标:选择准确度 (accuracy, ACC) 和 F1_score (F1) 衡量模型的检测性能。

2)对比方法:

①RED:直接通过筛选事件类型检索感兴趣事件,数据的时间跨度为 1 d,无须设置参数。

②FSM^[13]:一种利用频繁子图特征工程构造特征选择集并以此进行特征表示的二分类方法。正负例的最小支持度均设为 0.1, SVM 惩罚系数设为 $C = 1$,核函数设置为 $\text{kernel} = \text{'linear'}$,数据时

间跨度为 1 d,其余为默认设置。

③CompGCN^[17]:一种多关系图卷积神经网络,通过平均池化形成全局表示,最终输出事件概率。学习率 learning_rate 为 0.001,训练模型输入的批次大小 batch_size 为 10,节点与边嵌入的组合方式为循环相关 corr,丢弃率 dropout rate 为 0.3,采用 Adam 优化器^[21]更新模型参数,训练最大轮次 epoch 为 50,CompGCN 层数设置为 1 层、输入与输出维度均为 100,数据时间跨度为 1 d。

3)模型调优的目的是找到效果最佳的模块组合,具体为:

①TEFM_G:用双向门控循环单元 (bidirectional gated recurrent unit, GRU)挖掘时序演化特征。

②TEFM_L:用单向长短期记忆网络 (long short-term memory, LSTM)挖掘时序演化特征。

③TEFM:用 Bi_LSTM 挖掘时序演化特征。

4)参数设置:序列长度设为 9 d,其余设置同 CompGCN,代码运行的随机种子设置为 41 505。

5)实验环境:本文实验均在一台 CPU 配置为 Inter(R)Core (TM) i5 - 1135G7@2.40 GHz 的计算机上完成,运行的软件环境为 Windows 10 + Python 3.10 + Pytorch 1.11.0,运行的软硬件环境均一致。

3.3 对比实验

为验证所提方法性能,设计了对比实验。3 种基线方法用于测试模型性能,3 种时序特征学习器用于调优时序学习模块。实验方式为对每个数据集分别采用五折交叉验证的方法进行评估测试,指标结果为五次实验的平均值。实验结果 (包含五折实验的平均值与标准差)如表 3 所示。

表3 各模型在不同数据集上的检测性能

Tab.3 Detection performance of each model on different datasets

模型名称	US		IN		UKR	
	ACC	F1	ACC	F1	ACC	F1
RED	37.5 ± 5.4	53.8 ± 6.0	32.7 ± 3.6	48.9 ± 4.4	52.3 ± 7.2	64.3 ± 7.5
FSM	<u>91.3</u> ± 2.5	<u>87.9</u> ± 2.7	86.5 ± 1.6	78.7 ± 3.2	<u>83.0</u> ± 2.0	<u>80.0</u> ± 4.2
CompGCN	87.4 ± 4.1	82.6 ± 4.9	<u>86.5</u> ± 4.3	<u>79.7</u> ± 7.3	71.8 ± 1.9	67.3 ± 5.1
TEFM_G	96.7 ± 1.2	94.0 ± 4.0	94.8 ± 1.9	88.5 ± 9.0	91.5 ± 4.1	87.1 ± 6.4
TEFM_L	93.2 ± 5.3	91.5 ± 4.0	94.0 ± 4.9	90.4 ± 8.8	91.5 ± 4.9	87.7 ± 6.6
TEFM	96.4 ± 2.6	95.4 ± 3.0	96.5 ± 2.2	94.8 ± 3.6	88.2 ± 6.5	86.0 ± 9.3
相对提升	5.4	7.5	10.0	15.1	8.5	7.7

注:表中带下划线的数据为 3 组基线模型中表现最好的数据,加粗数据为 3 组调优时序学习模块中表现最好的数据。

纵向上看,TEFM 算法性能优于对比算法。RED 方法是通过对当日的实验数据检索实现事件检测,如当日存在抗议类的事件记录则判定事件发生,反之视为不发生,但因数据库获取结构化事件记录的过程存在一定的误差,易造成虚警率过高。FSM 算法通过其改进的频繁子图算法挖掘得到正负例频繁子图集(显式关联),通过取并集后去除交集得到特征选择集,并以此进行特征表示,实现事件检测。该算法在一定程度上降低了数据库的误差,但这种基于特征工程的方法是基于规则挖掘特征的,显式关联能够被有效挖掘,隐式关联无法挖掘,此外由于社会事件往往是演化发展的,时序演化因素也是一个需要考虑的重要影响因素。CompGCN 通过接入全局表示模块能够实现对多关系异构图的图分类任务,进而实现事件检测,在一定程度上挖掘到了隐式关联关系,提升了检测性能,但因没有通过注意力机制关注关键特征,没有考虑时序演化因素,检测能力不及所提算法。所提 TEFM 算法通过构造实体交互图降低了数据库自身的误差与冗余,通过改进的 CompGCN 网络实现显式与隐式关联的融合挖掘,通过基于注意力机制的图表示层得到较高质量的图全局表示,通过挖掘合适时间跨度的演化序列实现了最终的事件检测任务,解决了以往研究的不足。横向上看,TEFM 算法在 2019 至 2022 年间 6 个城市的数据上均表现出稳定、优异的效果,验证了所提方法的有效性。

另外,为了使模型具有更好的检测性能,本文设计了时序特征学习器的调优实验,分别基于 3 种典型的循环神经网络构建时序演化特征挖掘层。实验结果表明,3 种时序特征学习器相差不明显,这是因为在以往的研究中已经表明 LSTM 和 GRU 的性能近乎持平。Bi_LSTM 普遍比 LSTM 检测性能高一些,是由于得到了前向与后向两个方向的特征。Bi_LSTM 与 Bi_GRU 相比,检测性能总体上稍胜一筹,这是由于 Bi_LSTM 相比于 Bi_GRU 多了一个门控单元,拟合效果稍好。

3.4 泛化实验

为了检验泛化性能,本文在调优时序特征学习器(Bi_LSTM)的基础上,共设 3 组实验。由于数据集在时间上存在先后,即 UKR > IN > US,因此 3 对实验组分别为 US-IN、US-UKR、IN-UKR,其中每组前面的数据集为训练集,每组后面的数据集为测试集,参数设定与评价指标保持不变,结果如图 8 所示。

由实验结果可知,所提模型具有较强的泛化

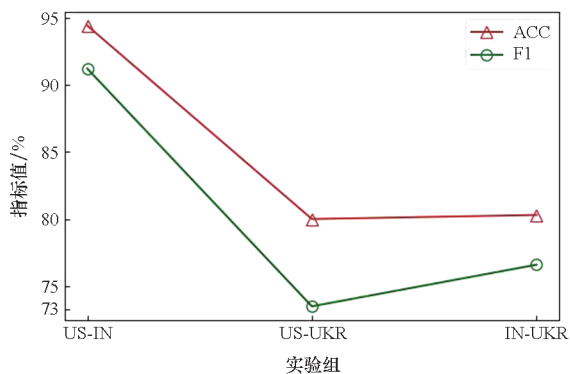


图 8 泛化分析实验

Fig. 8 Generalization analysis experiment

能力。第一,模型整体上表现较好,在各组实验中均取得较好的效果,反映出模型的泛化性较强。这说明社会事件在发展演化过程中存在相似性,验证了通过从历史数据挖掘实体交互与时序演化特征实施事件检测的可行性,这与此前研究结论一致。第二,US-UKR 与 IN-UKR 表现相对偏低,分析原因可知,相应实验组均包含 UKR。UKR 主要内容是围绕俄乌冲突在乌克兰首都基辅发生的抗议,而与之对应的训练集均为和平时期对政府或公共组织表达抗议,抗议内容存在偏差,导致抗议事件发生的交互模式存在一定的差别,进而导致方法的泛化效果有所降低。

3.5 参数敏感分析实验

参数敏感度以及消融等性能分析实验将使用不同的数据集进行训练与测试。这是由于性能分析的目的是检验模型在贴近真实场景下的性能,而真实场景通常在不同的数据集上进行训练与测试,因此使用不同的数据集进行训练与测试,以达实验目的。

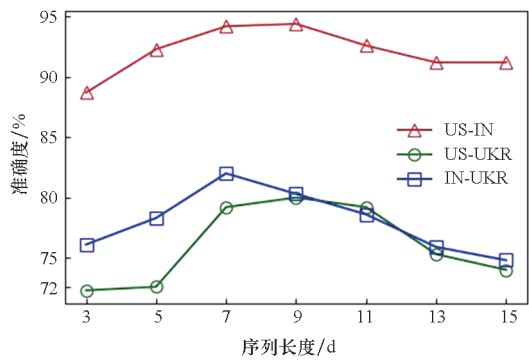
3.5.1 历史演化序列长度实验

本节将序列长度设置在 3 ~ 15 d 不等(间隔为 2 d),分组与参数设定同 3.4 节,结果如图 9 所示。

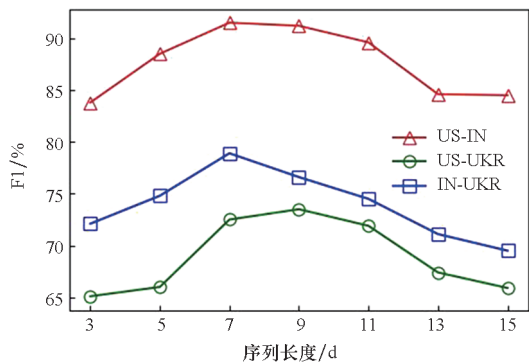
模型在序列长度为 7 ~ 11 d 的检测性能最佳。由于所提方法是基于历史演化序列挖掘的方法,理论上序列的长短会对检测性能产生一定影响。可以看到,序列长度过小会由于序列特征较少而导致特征学习不充分,进而导致检测性能下降;序列长度过大则会将距离事件发生时间较远的无关信息引入,影响模型对关键区分特征的学习,进而导致检测性能下降。

3.5.2 时序演化特征挖掘层层数实验

为验证时序演化特征挖掘层层数对检测性能



(a) 准确度
(a) Accuracy



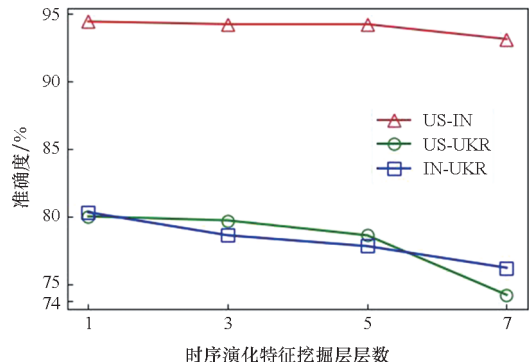
(b) F1 值
(b) F1 score

图9 序列长度分析实验

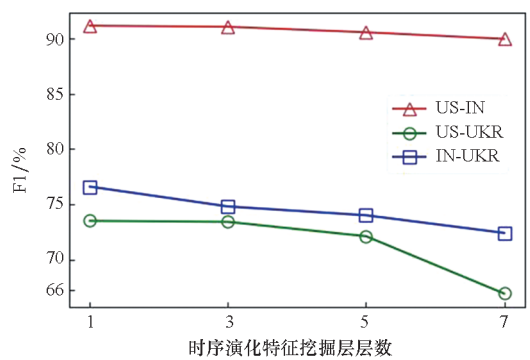
Fig.9 Sequence length analysis experiment

的影响,本节将 Bi_LSTM 的层数分别设置为 1~7 层不等(间隔为 2),共设 3 组实验,分组与参数设定情况同泛化实验,实验结果如图 10 所示。

实验结果表明,时序演化特征挖掘层层数对检测性能影响不大,实验组指标均表现出随层数增加而下降的趋势,总体上 1 层实验指标相对最佳。一方面,由于实验数据集的样本数量相对有限,因此通过加深层数并不能拟合得到更好的实验结果;另一方面,随着层数的增加,时序演化特征挖掘层的复杂度相对增加,由于数据样本相对



(a) 准确度
(a) Accuracy



(b) F1 值
(b) F1 score

图10 时序演化特征挖掘层层数实验

Fig.10 Layer experiment of temporal feature evolution mining layer

有限,因此过拟合的情况较容易出现,进而导致检测性能下降。所以,在实际运用中,可通过增加样本数量的方式,合理设置挖掘层的层数,以提升检测性能。

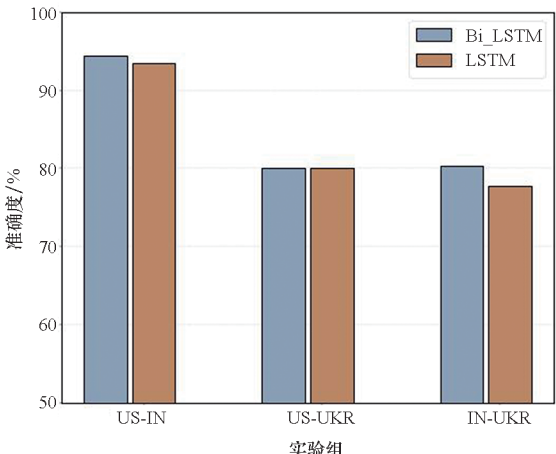
3.5.3 LSTM 网络单双向实验

为验证单向 LSTM 与双向 LSTM 对检测的影响,本节在 3 个实验组上分别对比了两种网络的检测性能,分组与参数设定情况同泛化实验,实验结果如图 11 所示。

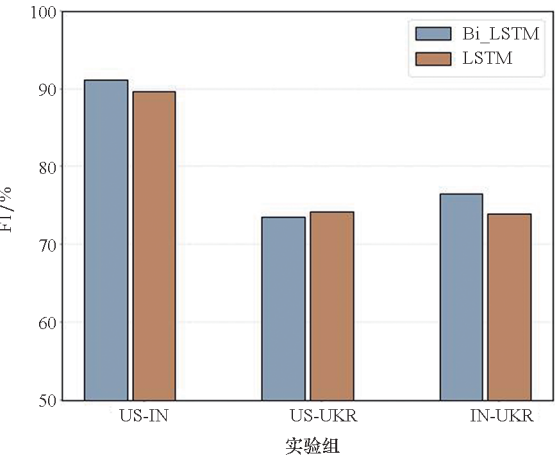
实验结果表明,双向 LSTM 网络总体上对模型的检测能力起到积极作用。双向 LSTM 网络能够从前后两个方向学习序列特征,捕获序列前端与后端决定检测性能的关键特征,有效弥补了单向 LSTM 网络仅能从前向学习序列的不足,因此模型的检测性能相对得到提升。

3.5.4 序列输出全局表示方式实验

为验证序列输出的全局表示方式对检测性能的影响,本节分别对 LSTM 的每个输出采取平均



(a) 准确度
(a) Accuracy

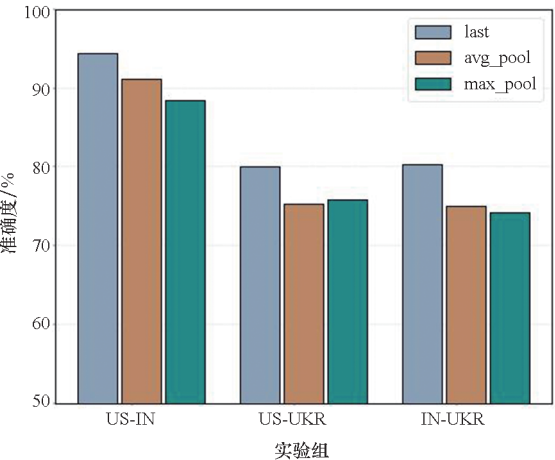


(b) F1 值
(b) F1 score

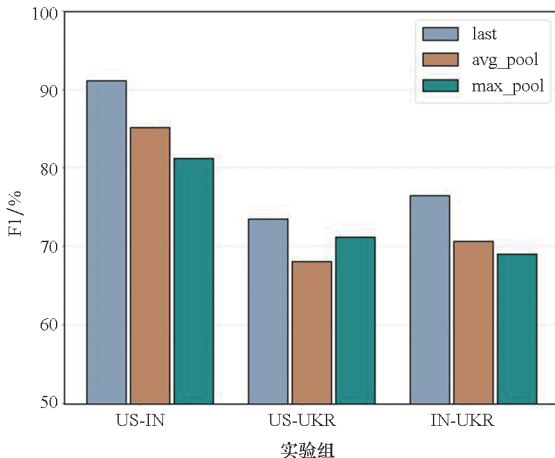
图 11 LSTM 单双向实验
Fig. 11 LSTM unidirectional and bidirectional experiment

池化、最大池化以及本文采用的序列末端输出 3 种方式实现序列的全局表示,分别记为 avg_pool、max_pool、last。实验依然为 3 组,分组与参数设定均与泛化实验一致,实验结果如图 12 所示。

实验结果表明,采用序列末端作为时序演化特征挖掘层输出的方式相较于其他两种方式优势明显。一方面,平均池化输出方式的原理是对 LSTM 各单元的等值加权输出,由于更新后的序列末端单元已经蕴含了整个序列信息,故平均池化输出的方式引入了过多的噪声信息,进而导致模型检测性能下降。另一方面,最大池化输出方式的原理是将 LSTM 各单元输出嵌入的最大值作为整个序列的输出,所得到的序列全局嵌入不能较好代表整个序列,而采用末端单元输出的方式则能够将序列历史信息充分捕捉,又不至于引入过多无关噪声,因此能够取得较好的检测效果。



(a) 准确度
(a) Accuracy



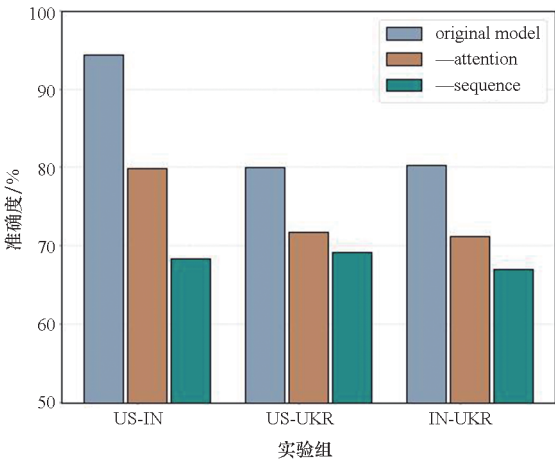
(b) F1 值
(b) F1 score

图 12 序列全局表示方式实验
Fig. 12 Experiment of sequence global embedding

3.6 消融分析实验

为了分析模块层的作用,本节在 3 对数据集上共设置 3 组消融分析实验。第 1 组为提出的原始模型,记为 original model;第 2 组为去除时序演化特征挖掘层,记为一sequence,做法是在去除时序演化特征挖掘层后,将原始模型图表示层的输出进行平均池化,得到整个序列的输出;第 3 组为去除注意力机制,记为一attention,做法是将原始模型图表示层的注意力部分替换为对节点与边的平均池化。实验结果如图 13 所示,结果表明,模型的相应组件均对检测性能具有积极作用。

一方面,去除时序演化特征挖掘层后,实验指标均有所下降。这是因为将时序演化特征挖掘层改为平均池化的方式不能有效捕获历史演化序列的特征,平均池化的方式也在一定程度上弱化了



(a) 准确度
(a) Accuracy

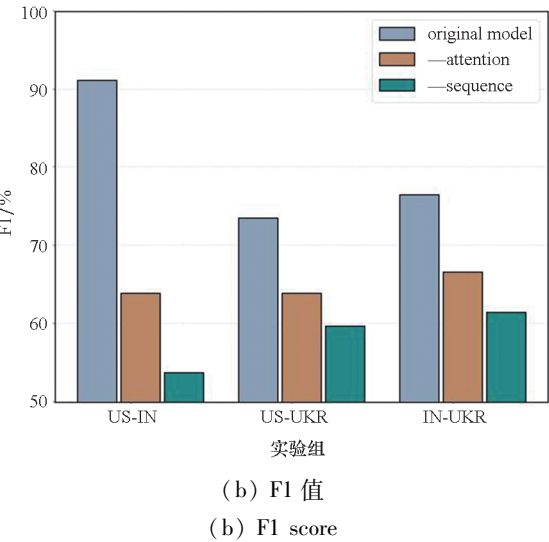


图 13 消融分析实验

Fig. 13 Ablation analysis experiment

决定事件发生的关键特征,而时序演化特征挖掘层能够将过去的重要信息通过网络传递到序列末端,得到的序列全局表示具有较好的代表性,因此检测性能较好。另一方面,去除注意力机制后,实验指标也均有不同程度的降低。这是因为注意力机制能够为序列单元中决定事件发生的关键节点与边分配更高的权重,进而得到更具代表性的全局表示,因此检测效果更佳。

4 结论

本文将特定域社会事件检测任务建模为有监督的二分类问题,构建了检测模型,主要包括交互特征挖掘层、图表示层、时序演化特征挖掘层与输出层。通过构造实体交互图,降低了数据库误差影响;通过改进的 CompGCN 网络挖掘序列单元的图结构信息,利用注意力机制关注重要特征从而得到序列单元的高质量全局表示,解决了显式与隐式关联融合挖掘的问题;利用循环神经网络提取时序演化特征,解决了时序演化因素挖掘的问题。实验结果表明,所提方法能够有效检测社会事件。

未来将主要关注两方面工作:一是考虑在线检测,将现有模型部署到现实系统,设置多个事件类型检测专题,实现多类型感兴趣事件的在线检测;二是缩小用于构造序列单元的时间跨度,目前 GDELT 的更新频率为 15 min,探索将时间跨度缩小至 15 min 并在此基础上合理设置序列长度,能够进一步提升检测性能。

参考文献 (References)

[1] AFYOUNI I, KHAN A, AGHBARI Z A. Deep-Eware;

spatio-temporal social event detection using a hybrid learning model[J]. Journal of Big Data, 2022, 9(1): 86.

[2] ATEFEH F, KHREICH W. A survey of techniques for event detection in Twitter[J]. Computational Intelligence, 2015, 31(1): 132–164.

[3] AFYOUNI I, AL AGHBARI Z, RAZACK R A. Multi-feature, multi-modal, and multi-source social event detection: a comprehensive survey[J]. Information Fusion, 2022, 79: 279–308.

[4] MREDULA M S, DEY N, RAHMAN M S, et al. A review on the trends in event detection by analyzing social media platforms' data[J]. Sensors (Basel), 2022, 22(12): 4531.

[5] ARAKI J, MITAMURA T. Open-domain event detection using distant supervision [C]//Proceedings of the 27th International Conference on Computational Linguistics, 2018.

[6] SAKAKI T, OKAZAKI M, MATSUO Y. Earthquake shakes Twitter users: real-time event detection by social sensors[C]//Proceedings of the 19th International Conference on World Wide Web, 2010: 851–860.

[7] LEE K, QADIR A, HASAN S A, et al. Adverse drug event detection in tweets with semi-supervised convolutional neural networks [C]//Proceedings of the 26th International Conference on World Wide Web, 2017: 705–714.

[8] 蒋沂桔. 基于图卷积网络的事件检测及预测技术研究与实现[D]. 长沙: 国防科技大学, 2021.

JIANG Y J. Research and implementation of event detection and prediction technology based on graph convolutional network [D]. Changsha: National University of Defense Technology, 2021. (in Chinese)

[9] CARTA S, CONSOLI S, PIRAS L, et al. Event detection in finance using hierarchical clustering algorithms on news and tweets[J]. PeerJ Computer Science, 2021, 7: e438.

[10] LEETARU K, SCHRODT P A. GDELT: global data on events, location, and tone, 1979–2012 [C]//Proceedings of ISA Annual Convention, 2013.

[11] O'BRIEN S P. Crisis early warning and decision support: contemporary approaches and thoughts on future research[J]. International Studies Review, 2010, 12(1): 87–104.

[12] KENESHLOO Y, CADENA J, KORKMAZ G, et al. Detecting and forecasting domestic political crises: a graph-based approach[C]//Proceedings of the ACM Conference on Web Science, 2014: 192–196.

[13] QIAO F C, LI P, DENG J S, et al. Graph-based method for detecting occupy protest events using GDELT dataset [C]//Proceedings of the International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, 2015: 164–168.

[14] AZEEMI A H, SOHAIL M H, ZUBAIR T, et al. RevDet: robust and memory efficient event detection and tracking in large news feeds[C]//Proceedings of Advanced Analytics and Learning on Temporal Data, 2021: 170–185.

[15] VELIKOVI P, CUCURULL G, CASANOVA A, et al. Graph attention networks [C]//Proceedings of the 6th International Conference on Learning Representations, 2018: 1–12.

[16] SCHLICHTKRULL M, KIPF T N, BLOEM P, et al.

Modeling relational data with graph convolutional networks[C]// Proceedings of European Semantic Web Conference, 2018: 593 – 607.

[17] VASHISHTH S, SANYAL S, NITIN V, et al. Composition-based multi-relational graph convolutional networks [C]// Proceedings of the 8th International Conference on Learning Representations, 2020: 1 – 15.

[18] GLOROT X, BENGIO Y. Understanding the difficulty of training deep feedforward neural networks [J]. Journal of Machine Learning Research, 2010, 9: 249 – 256.

[19] 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, 2017: 6000 – 6010.

[20] 刘建伟, 宋志妍. 循环神经网络研究综述[J]. 控制与决策, 2022, 37(11): 2753 – 2768.

LIU J W, SONG Z Y. Overview of recurrent neural networks[J]. Control and Decision, 2022, 37(11): 2753 – 2768. (in Chinese)

[21] KINGMA D P, BA J. Adam: a method for stochastic optimization [C]//Proceedings of the 3rd International Conference for Learning Representations, 2015.