

GCAN: Graph-aware Co-Attention Networks for Explainable Fake News Detection on Social Media

关键词：假新闻检测、图卷积网络

Keywords: Fake News Detection, Graph Convolutional Network

问题定义

这篇文章是基于推特的推文做假新闻检测，通过推文本身的文本、转发用户的信息、转发行为作为输入数据检测，基于有监督训练学习假新闻检测。

假新闻检测首先具有的数据是新闻文本 $\Psi = \{s_1, s_2 \dots s_{|\Psi|}\}$ 其中 $s_i = \{q_1^i, q_2^i \dots q_{l_i}^i\}$ 表示文本的词序列。除此之外还有用户集合 $U = \{u_1, u_2 \dots u_{|U|}\}$ ，推文 i 被转发的集合表示为 $R_i = \{\dots (u_j, \mathbf{x}_j, t_j) \dots\}$ ，其中 u_j 为转发的用户， \mathbf{x}_j 是它的用户特征表示， t_j 表示转发时间。模型基于推文和推文相关的信息做一个二分类任务。

文章简述

这篇文章个人认为两个亮点：

- 一个是构建了用户-用户的图网络，作者认为转发真新闻的用户和转发假新闻的人在人群特征上是存在差异的，所以利用到了GNN的聚类性，在用户图上跑GCN获得传播后的用户表示。
- 一个是引入Co-Attention，让待检测的推文的词序列和转发它的用户做Co-Attn，以捕捉哪类用户关注新闻的哪些部分，以提供一定的解释性。

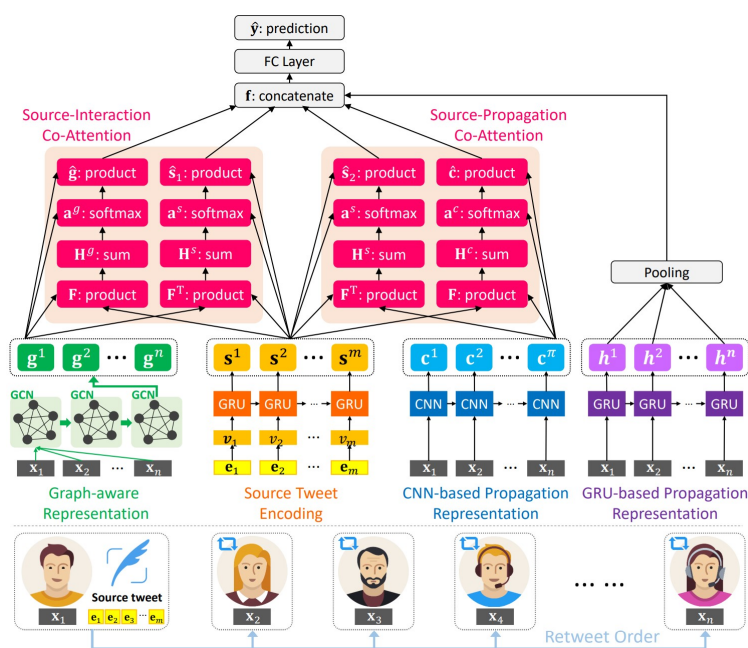


Figure 1: The architecture of our GCAN model.

详解

用户建模

建模推文的转发者对识别推文真实性的意义是很大的，作者这里用了如下几种属性：

1. number of words in a user's self-description
2. number of words in u_j 's screen name
3. number of users who follows u_j
4. number of users that u_j is following
5. number of created stories for u_j
6. time elapsed after u_j 's first story
7. whether the u_j account is verified or not
8. whether u_j allows the geo-spatial positioning
9. time difference between the source tweet's post time and u_j 's retweet time, and
10. the length of retweet path between u_j and the source tweet (1 if u_j retweets the source tweet).

推文编码

这里作者比较简单，直接对Embedding做一个非线性映射，并使用GRU得到推文编码。

$$\mathbf{S} = [\mathbf{s}^1, \mathbf{s}^2, \dots, \mathbf{s}^m] \in \mathbb{R}^{d \times m}$$

其中 m 是单词数， d 是特征维度。

序列上的转发者传播表示

这里作者考虑对推文的一系列转发者 $R_i = \{\dots(u_j, \mathbf{x}_j, t_j)\dots\}$ 进行建模，在它们之间传播信息。为了提高用户特征的质量，如果转发者足够多则选择时间最新的前 n 个，不足则重采样到 n 个，得到 $\langle \dots, \mathbf{x}_t, \dots \rangle, t \in \{1, \dots, n\}$ 。

其提出两个方法：

1. GRU编码，对序列跑一边GRU，并均值池化。
2. CNN编码，用1D卷积对序列编码。

图上的转发者传播表示

作者希望用图建模用户间的交互关系，但由于数据未给出用户之间的交互，所以作者为每一推文的转发者建了一张全联通图，并使用用户间特征向量的相似度作为权重。在图上做GCN，得到图的表示

$$\mathbf{G} \in \mathbb{R}^{g \times n}。$$

Dual Co-Attention

作者在建模推文和用户后，希望构建两者间的交互性表示，以挖掘何类用户关注推文的何种部份。

具体而言，作者让推文表示分别与序列的CNN表示以及图上表示进行Co-Attention（都是序列表示，所以不详细展开怎么做了），这里不用GRU的原因主要是GRU的均值操作抹去了用户个体的独立性（不过并不理解为什么不能省略均值池化，这样也可以以用户为单位做Co-Attn啊，比CNN细粒度还高一点，可能这部分是作者实验效果不好，编了一段吧）。

与跨模态的Co-Attention不同，这里没有跨View的特征融合，直接得到attention矩阵对输入特征矩阵聚合得到特征向量。

答案预测

最终用上面得到的各种表示生成答案。

总结

这篇文章的思路还是比较新的，Co-Attention真的在各种地方都可以试试看，GCN部分的提出的idea还是不错的，认为转发假新闻的用户是聚类的，所以引入了GCN，但这篇文章只对某篇推文的用户建了子图，但我觉得像节点分类一样，应该用所有的用户建图，用用户-推文-用户以及关注等interaction构造图表示，可能会有效果。