DOI: 10.11992/tis.201809037
网络出版地址: http://kns.cnki.net/kcms/detail/23.1538.TP.20190401.0833.002.html

引入外部词向量的文本信息网络表示学习

张潇鲲, 刘琰, 陈静
(数学工程与先进计算国家重点实验室, 河南 郑州 450000)

摘 要: 针对信息网络(text-based information network)现有研究多基于网络自身信息建模, 受限于任务语料规模, 只使用任务相关文本进行建模容易产生语义漂移或语义残缺的问题, 本文将外部语料引入建模过程中, 利用外部语料得到的词向量对建模过程进行优化, 提出基于外部词向量的网络表示模型 NE-EWV(network embedding based on external word vectors), 从语义特征空间以及结构特征空间两个角度学习特征融合的网络表示。通过实验, 在现实网络数据集中对模型有效性进行了验证。实验结果表明, 在链接预测任务中的 AUC 指标, 相比只考虑结构特征的模型提升 7%~19%, 相比考虑结构与文本特征的模型在大部分情况下有 1%~12% 提升; 在节点分类任务中, 与基线方法中性能最好的 CANE 性能相当。证明引入外部词向量作为外部知识能够有效提升网络表示能力。

关键词: 网络表示学习; 文本信息网络; 自编码器; 外部词向量; 节点分类; 词向量; 分布式表示; 表示学习
中图分类号: TP181 **文献标志码**: A **文章编号**: 1673−4785(2019)05−1056−08

中文引用格式: 张潇鲲, 刘琰, 陈静. 引入外部词向量的文本信息网络表示学习 [J]. 智能系统学报, 2019, 14(5): 1056–1063.
英文引用格式: ZHANG Xiaokun, LIU Yan, CHEN Jing. Representation learning using network embedding based on external word vectors[J]. CAAI transactions on intelligent systems, 2019, 14(5): 1056–1063.

Representation learning using network embedding based on external word vectors

ZHANG Xiaokun, LIU Yan, CHEN Jing
(Mathematical Engineering and Advanced Computing State Key Laboratory, Zhengzhou 450000)

Abstract: Network embedding, which preserves a network's sophisticated features, can effectively learn the low-dimensional embedding of vertices in order to lower the computing and storage costs. Content information networks (such as Twitter), which contain rich text information, are commonly used in daily life. Most studies on content information network are based on the information of the network itself. Distributed word vectors are becoming increasingly popular in natural language processing tasks. As a low-dimensional representation of the semantic feature space, word vectors can preserve syntactic and semantic regularities. By introducing external word vectors into the modeling process, we can use the external syntactic and semantic features. Hence, in this paper, we propose network embedding based on external word vectors (NE-EWV), whereby the feature fusion representation is learned from both semantic feature space as well as structural feature space. Empirical experiments were conducted using real-world content information network datasets to validate the effectiveness of the model. The results show that in link prediction task, the AUC of the model was 7% to 19% higher than that of the model that considers only the structural features, and in most cases was 1% to 12% higher than the model that considers structural and text features. In node classification tasks, the performance is comparable with that of context-aware network embedding (CANE), which was the state-of-the-art baseline model.

Keywords: network embedding; content information network; auto-encoder; external word vectors; vertex classification; word vectors; distributed representation; representation learning

收稿日期: 2019−09−19. **网络出版日期**: 2019−04−02.
基金项目: 国家自然科学基金项目 (61309007, U1636219); 国家重点研发计划课题资助项目 (2016YFB0801303, 2016QY01W0105).
通信作者: 刘琰. E-mail: ms_liuyan@aliyun.com.

近年来, 随着互联网的发展, 以 Facebook、twitter、微博等为代表的大型网络不断发展, 产生了海量具有网络结构的数据, 这些数据的特点在于样本

点之间并不完全独立, 而是具有一定的连接关系, 同时网络节点自身也包含特定的属性信息。日常生活中的社交网络 (微博)、问答社区 (知乎)、生活服务类网站 (大众点评)、论文引用关系网络等包含了大量文本信息, 下文中将此种网络简称为文本信息网络。在文本信息网络中, 文本信息以标签、正文、描述以及其他元数据形式广泛存在, 给网络提供了大量可利用的语义信息。例如论文引用关系网络中, 论文作为网络节点并以引用关系作边, 节点还包含相关文本信息。网络数据的这些特性, 给大规模或复杂网络数据研究带来了挑战。

网络表示学习 (network embedding 或 network representation learning) 目的是学习网络节点的低维空间向量表示, 降低存储、计算成本, 提升并行能力, 使传统机器学习算法能够在大规模数据中得到应用[1]。因此, 近年涌现出许多相关研究, 其研究成果在链接预测[2]、社团发现[3]、节点分类[4]、相似度计算[5]、网络可视化[6] 等应用场景广泛应用。大部分已有网络表示学习算法基于网络本身特征进行表示学习, 例如 DeepWalk[7]、Node2Vec[8]、Line[9] 等刻画结构特征的模型; 以及针对文本信息网络, 在 DeepWalk[7] 基础上引入文本特征的 TADW[10], 引入互注意力机制, 并在部分文本信息网络公开数据集中得到了目前最优结果的 CANE[11]。文本信息网络表示现有方法从网络本身文本特征出发, 由于网络文本分布与自然语言文本分布差异, 会产生一定程度的语义残缺或语义漂移, 这种情况在数据集规模受限情况下更为明显。

直觉上, 为模型引入越多外部知识, 模型的表示容量越高, 模型结果越能够刻画更多网络特征; 而预训练的分布式词向量正是针对文本相关任务的外部语义知识。随着词向量应用的普及, 存在许多以通用语料训练得到的词向量资源, 其中包含了大量语义信息。利用这部分已有语义资源增强文本信息网络的表示是本文研究的目标。

1　相关工作

网络表示学习早期技术以图表示 (graph embedding)、降维方法为主。包括 multidimensional scaling (MDS)[12]、IsoMap[13]、局部线性表示 LLE[1] 以及 Laplacian Eigenmap[14]。这类算法的计算复杂度偏高, 不适合在大规模网络中应用。

随着近年网络表示学习发展, 大量可以应用在大规模网络中的算法相继提出。对于文本信息网络, 主要分为如下 2 类:

1) 只考虑结构特征的网络表示学习方法

Deepwalk[7] 作为网络表示学习的经典算法, 将自然语言处理中利用词共现信息进行建模的算法 SkipGram[1] 引入到网络表示学习任务中, 通过随机游走构建节点上下文序列, 并利用 Hierarchical Softmax[2] 的树形结构加速训练过程。LINE[8] 主要利用预先设计的概率密度函数来表征图的一阶、二阶相似度, 并引入负采样[1]、异步随机梯度下降 (ASGD)[15] 降低计算复杂度, 实现适用于大规模网络节点表示的计算。Node2vec[9] 对 Deepwalk 的随机游走策略进行了修改, 通过在游走路径中增加权重项来控制深度 (DFS) 以及广度 (BFS) 优先的游走方式, 使算法的图游走策略更有效率。GraRep[16] 将 k 阶相似矩阵进行分解, 并将得到的特征向量进行拼接得到最后的节点向量, 以此来捕捉更高阶的相似度特征, 但面临着计算量巨大的问题。网络结构的相似性主要体现在相似度计算上, 其中一阶、二阶相似度是最普遍使用的特征, 一般来说, 模型中包含越多的高阶相似度特征, 模型表现越好, 但是相应计算量也会增大。

2) 结合节点语义信息的网络表示学习方法

上述模型只考虑网络的结构特征信息, 针对文本信息网络, Yang 等[10] 提出了 text-associated Deep-Walk (TADW), 将文本信息与 DeepWalk 算法进行了结合。Tu 等[17] 提出了 max-margin DeepWalk (MMDW), 利用 SVM 思想对 DeepWalk 在文本信息网络中的应用进行改进, Tu 等[11] 提出了上下文相关的网络表示学习模型 CANE, 针对不同上下文, 利用互注意力机制, 学习网络节点在不同上下文中的表示。

使用自身文本特征进行建模, 受限于任务本身语料, 容易产生语义偏差或残缺。在论文写作时所知, 鲜见引入外部词向量辅助文本信息网络建模的研究。

2　语义漂移现象

如表 1 所示, 采用 Word2vec[1] 对实验部分的 Zhihu 数据集[12] 训练词向量, 对由训练得到的词向量与外部词向量中的随机词的相似词进行了对比。在 Zhihu 数据集词表中随机抽取两个词 “电子乐”、“杭州”, 根据余弦相似度分别在 Zhihu 词向量与外部词向量词表中找到前 5 个表示近似的词。可以看到, 受限于数据集规模, Zhihu 数据集的词模型表示能力有限, 语义漂移明显。

表 1 “电子乐”相似词对比
Table 1 “Dian Zi Yue” cosine similarity

电子乐(Zhihu)	相似度	电子乐(外部词向量)	相似度
专指	0.828	电音	0.717
energy	0.801	电子乐	0.710
sound	0.782	爵士音乐	0.709 1
录音	0.747	音乐风格	0.705 6
现场表演	0.740	乐队	0.702 4

表 2 “杭州”相似词对比
Table 2 “Hang zhou” cosine similarity

杭州(Zhihu)	相似度	杭州(外部词向量)	相似度
1897	0.949	嘉兴	0.765
人间天堂	0.906	无锡	0.744
浙大	0.866	苏州	0.744
文化名城	0.795	宁波	0.699
坐落于	0.750	上海	0.692

3 问题定义与描述

沿用 LINE[9] 中的信息网络定义，文本信息网络定义如下：

定义 1 文本信息网络

文本信息网络被定义为 $\boldsymbol{G}=(\boldsymbol{V},\boldsymbol{E},\boldsymbol{T})$，$\boldsymbol{V}$ 表示网络中的节点集合，$\boldsymbol{T}$ 表示节点文本信息集合，$\boldsymbol{E}\subseteq\boldsymbol{V}\times\boldsymbol{V}$ 表示节点通联构成的边，$e=(u,v)$、$e\in\boldsymbol{E}$ 代表了节点 u、v 之间有带有权重 $w_{uv}>0$ 的边。对于无向图 $w_{uv}\equiv w_{vu}$、$(u,v)\equiv(v,u)$，对于有向图 $w_{uv}\neq w_{vu}$、$(u,v)\neq(v,u)$。

定义 2 引入外部词向量的文本信息网络

给定外部词向量 $\boldsymbol{C}$，$\boldsymbol{C}$ 为在语义特征空间中的词向量表示集合。此时文本信息网络以 $\boldsymbol{G}=(\boldsymbol{V},\boldsymbol{E},\boldsymbol{T},\boldsymbol{C})$ 定义。

定义 3 节点特征空间表示

网络表示学习的目的是对每个节点 $v\in\boldsymbol{V}$ 学习一个低维空间的向量表示 $\boldsymbol{v}\in\mathbf{R}^d(d<<|\boldsymbol{V}|)$。

对于节点 $v\in\boldsymbol{V}$，其节点文本表示为 $\text{content}_v=(\text{word}_1,\text{word}_2,\cdots,\text{word}_{N_v})$，节点特征空间表示即节点在不同向量特征空间中的表示结果。在文本信息网络中，特征空间主要分为描述文本语义特征的语义特征空间，节点 v 在其中表示记为 $\boldsymbol{v}^t$，此时节点与词在同一特征空间中，能够以节点文本语义衡量节点之间的相似程度；以及描述网络结构特征的结构特征空间，节点 v 在其中表示记为 $\boldsymbol{v}^s$，词和节点的表示都在一定程度上包含了网络结构相关特征。

定义 4 结构相似度

一阶相似度 一阶相似度通过当前节点与相邻节点间的联通关系，描述了网络在一跳范围内的结构特征。对节点 u、v，若节点间没有边相连，则一阶相似度为 0。若存在边 (u,v)，一阶相似度即为边权重 w_{uv}。

二阶相似度 二阶相似度衡量了当前节点与相距两跳的邻居节点间的结构相似程度。记 $p_u^{(1)}=(w_{u,1},w_{u,2},\cdots,w_{u,|V|})$ 为节点 u 与其他所有点之间的一阶相似度。u、v 的二阶相似度为 $p_u^{(1)}$、$p_v^{(1)}$ 的相似度，该相似度可以通过余弦相似度等相似度度量方式进行衡量，若 u、v 没有一跳公共邻居节点，则二阶相似度为 0。

高阶相似度 记 u 与其他所有节点的 $N-1$ 阶相似度为 $p_u^{(N-1)}$。可以依以上定义 u、v 的 N 阶相似度为 $p_u^{(N-1)}$ 与 $p_v^{(N-1)}$ 的相似程度，若 u、v 没有 $N-1$ 跳公共邻居节点，则 N 阶相似度为 0。

基于外部词向量的文本信息网络表示学习目的是对给定文本信息网络 $\boldsymbol{G}=(\boldsymbol{V},\boldsymbol{E},\boldsymbol{T},\boldsymbol{C})$，在融合结构特征与语义特征的特征空间中，学习网络节点的低维向量表示 $\boldsymbol{v}\in\mathbf{R}^d$，使表示结果包含网络结构特征、网络本身文本特征以及外部文本特征。出于计算复杂度考虑，本文只使用一阶、二阶相似度对结构特征进行建模，语义特征使用词向量信息进行建模。

4 基于外部词向量的文本信息网络表示学习模型 NE-EWV

文本信息网络建模过程中涉及到两个向量特征空间：语义特征空间、结构特征空间。受限于任务本身语料规模与词分布，文本信息网络建模得到的语义特征空间表示与实际语义会产生一定程度偏差。本文引入外部词向量作为先验知识辅助建模过程，可以扩展语义特征空间表示容量，修正部分语义误差。故 NE-EWV 主要解决 2 个问题，一是引入外部词向量信息对语义特征进行扩充；其次是学习融合结构特征、语义特征的表示结果。

4.1 NE-EWV 模型基本架构

NE-EWV 分为 3 个部分，NE-EWV1 在语义特征空间中引入结构特征约束，得到语义特征空间中包含部分结构特征约束的节点表示 $\boldsymbol{v}^t$。NE-EWV2 在结构特征空间中引入语义特征约束，得到结构特征空间中包含部分语义特征约束的节点表示 $\boldsymbol{v}^s$。NE-EWV3 表示结果由上述 2 步得到的

节点表示融合得到, 本文采用 2 种融合方式: 1) 简单将 2 个向量表示进行连接, 得到节点表示 $\boldsymbol{v}=\boldsymbol{v}^t\oplus\boldsymbol{v}^s$, 其中 $\oplus$ 代表向量拼接操作; 2) 基于自编码器的融合模型。

4.2 结构约束的语义特征空间表示模型 NE-EWV1

对于节点 $v\in\boldsymbol{V}$, 其节点文本 $\text{content}_v=(\text{word}_1,\text{word}_2,\cdots,\text{word}_{N_v})$, N_v 为节点 v 文本的词个数。节点外部词向量 $\boldsymbol{C}$, 若 word_i 在 $\boldsymbol{C}$ 存在, 记 word_i 在 $\boldsymbol{C}$ 中的词向量表示为 $\boldsymbol{C}_i$, 记 $\boldsymbol{M}_v$ 为 content_v 中词在 $\boldsymbol{C}$ 中出现的个数。则节点在语义特征空间中可以由词序列 $(\boldsymbol{C}_1,\boldsymbol{C}_2,\cdots,\boldsymbol{C}_{M_v})$ 唯一表示。

节点在语义特征空间中的表示受当前节点文本影响, 即节点的语义可以看作是节点文本中数个关键词的语义组合, 本文为简化起见, 对节点文本语义表示采用线性加权, 得到结果作为节点的语义表示。在实验章节第 5 节, 对 NE-EWV1 的可视化结果做了分析。

NE-EWV1 以节点文本词向量的线性加权作为节点在语义特征空间中的表示

$$\boldsymbol{v}^t=w_{v1}\boldsymbol{C}_1+w_{v2}\boldsymbol{C}_2+\cdots+w_{vM_v}\boldsymbol{C}_{M_v}\tag{1}$$

将表示限制在语义特征空间后, 沿用 LINE[9] 对于结构特征损失函数定义引入结构特征约束, 将问题转化为最优化问题求解。其中, 对于节点 u, v, 一阶相似度损失函数定义为

$$O_1=-\sum_{(u,v)\in E}w_{uv}\log p_1(\boldsymbol{u}^t,\boldsymbol{v}^t)\tag{2}$$

$$p_1(u^t,v^t)=\frac{1}{1+\exp(-\boldsymbol{u}^t\cdot\boldsymbol{v}^t)}\tag{3}$$

二阶相似度损失函数定义为

$$O_2=-\sum_{(u,v)\in E}w_{uv}\log p_2(\boldsymbol{u}^t|\boldsymbol{v}^t)\tag{4}$$

$$p_2(u^t|v^t)=\frac{\exp(\boldsymbol{u}^t\cdot\boldsymbol{v}^t)}{\sum\limits_{K=1}^{|V|}\exp(\boldsymbol{u}^t\cdot\boldsymbol{v}^t)}\tag{5}$$

对于表示结果, 沿用 LINE[9] 中对于一阶、二阶相似度的处理。由于一阶相似度只能应用于无向图, 对于有向图, 以二阶相似度作为结构特征的约束进行计算。对于无向图, 由一阶相似度损失函数得到节点表示记为 $\boldsymbol{v}^{t(1)}$, 二阶相似度损失函数得到节点表示记为 $\boldsymbol{v}^{t(2)}$, 通过向量拼接得到最后的语义特征空间表示 $\boldsymbol{v}^t=\boldsymbol{v}^{t(2)}\oplus\boldsymbol{v}^{t(2)}$。

4.3 结构约束的语义特征空间表示模型 NE-EWV2

为了引入语义约束, 将词看做特殊的网络节点, 以词向量相似度做权重边, 扩展原网络。考虑到模型计算量, 对每个节点 v 在外部词向量 $\boldsymbol{C}$ 中的节点文本, 首先通过采样得到节点文本的子集 $\text{content}_v'\subseteq\text{content}_v^C\subseteq\text{content}_v$, content_v^C 表示出现在外部词向量 $\boldsymbol{C}$ 以及节点文本 content_v 中的词集合; 对每个 $\text{word}_v\in\text{content}_v'$, 依次与 $\boldsymbol{V}$ 中其余节点文本的采样子集中每个词 $\text{word}_u\in\text{content}_u'$ 做边, 词向量的余弦相似度 $\text{sim}(\text{word}_v,\text{word}_u)$ 作为边的权重。对于有向图, $(\text{word}_v,\text{word}_u)=(\text{word}_u,\text{word}_v)$, $w_{uv}=w_{vu}=\text{sim}(\text{word}_v,\text{word}_u)$。

考虑到计算量因素, 引入 2 个概率变量 p、q, 其中 p 控制当前节点中词进行边扩展概率, q 控制目标节点中词进行边扩展概率。沿用 PageRank[18] 中对跳转概率的定义, 节点 v 的文本子集 content_v^C 中每个词以概率 $pr_1=(d_v/d_{\max})(1-p)+p$ 作为起始点做扩展边, 其中 d_v 为当前节点出度, $d_{\max}$ 当前网络中出度最大值; 对于目标节点 u 的文本子集 content_u^C, 其中每个词以概率 $pr_2=(d_u/d_{\max})(1-q)+q$ 作为扩展边的终点。

完成扩展网络后, 接下来在结构特征空间中与 4.2 节处理相同, 沿用 LINE[9] 中对损失函数的定义。将文本中的词与网络中的节点统一到结构特征空间中进行计算, 得到节点语义约束下的结构特征空间表示 $\boldsymbol{v}^s$。

4.4 表示融合模型 NE-EWV3

NE-EWV1、NE-EWV2 在不同程度上都包含了语义特征信息以及结构特征信息, 但建模过程侧重不同, 其表示结果属于不同特征空间。总的来说, NE-EWV1、NE-EWV2 描述了同一网络在不同视角下的网络表示, 对其表示结果做非线性变化映射到同一向量空间中, 其表示应当相对接近, 并可互为补充。因此文本提出 NE-EWV3 对 NE-EWV1、NE-EWV2 表示结果进行融合。

1) 通过向量拼接, 得到最终表示 $\boldsymbol{v}=\boldsymbol{v}^t\oplus\boldsymbol{v}^s$, 计算成本低。

2) 采用基于自编码器的表示融合模型。由于 $\boldsymbol{v}^s$、$\boldsymbol{v}^t$ 都包括了结构特征以及语义特征, 但侧重不同, 本文希望通过自编码器的非线性变换将 $\boldsymbol{v}^s$、$\boldsymbol{v}^t$ 映射到同一特征空间中, 相比方式 1), 由于采用了非线性变化, 模型理论上的表示能力更强。

由于在 2 个特征空间对同一事物进行表示, 当 $\boldsymbol{v}^s$、$\boldsymbol{v}^t$ 映射到同一向量空间中时, 其距离应当较为接近。NE-EWV3(AutoEncoder) 利用损失函数对 x_i、$\hat{x}_i$, y_i、$\hat{y}_i$, $\boldsymbol{v}^s$、$\boldsymbol{v}^t$ 的相似性进行约束, 得到最终的表示 $\boldsymbol{v}$。基于自编码器的表示融合模型如图 1 所示。

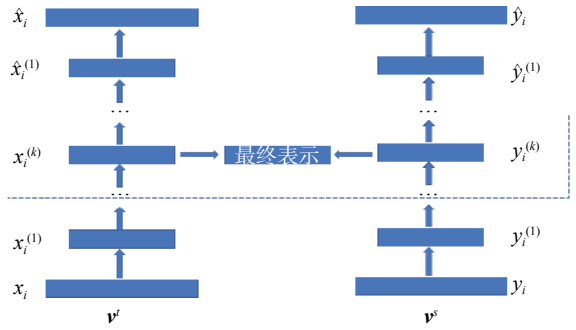


图 1　基于自编码器的表示融合模型

Fig. 1　Feature fusion representation model based on aligned auto-encoder

自编码器主要包括编码和解码 2 个过程，编码过程将输入映射到目标向量空间中，解码过程将目标向量空间中的表示还原到原输入向量空间中，要使目标向量空间的表示有效，需要解码过程中重建到输入向量空间中的表示与输入表示尽可能一致。

NE-EWV3(AutoEncoder) 采用了对称的自编码器结构，学习 $\boldsymbol{v}^s$、$\boldsymbol{v}^t$ 在目标特征空间中的表示结果。模型左右计算流程一致，这里以左侧为例进行说明。左侧自编码器的目的是将节点在语义特征空间中的表示 $\boldsymbol{v}^t$ 进行非线性变换。模型左侧初始输入为节点在语义特征空间中的表示 $\boldsymbol{v}^t$，编码阶段，通过下式得到压缩表示 $x_i^{(k)}$：

$$x_i^{(1)} = \sigma(W_x^{(1)} x_i + b^{(1)}) \tag{6}$$

$$x_i^{(k)} = \sigma(W_x^{(k)} x_i^{(k-1)} + b^{(k)}) \tag{7}$$

式中：σ 为 sigmoid 激活函数；$\sigma(x) = 1/(1+\exp(-x))$。

在解码阶段：

$$\hat{x}_i^{(k-1)} = \sigma(\hat{W}_x^{(k-1)} \hat{x}_i^{(k)} + \hat{b}^{(k-1)}) \tag{8}$$

$$\hat{x}_i = \sigma(\hat{W}_x^{(1)} \hat{x}_i^{(1)} + \hat{b}^{(1)}) \tag{9}$$

得到还原后的表示 $\hat{x}_i$。

左侧自编码器的损失函数定义为 $L_1 = \|x_i - \hat{x}_i\|_2^2$，右侧同理定义损失函数为 $L_2 = \|y_i - \hat{y}_i\|_2^2$。为了使 $\boldsymbol{v}^s$、$\boldsymbol{v}^t$ 能够被尽可能压缩到同一特征空间，需要压缩表示 $x_i^{(k)}$ 与 $y_i^{(k)}$ 足够接近，因此定义损失函数 $L_3 = \left\|x_i^{(k)} - y_i^{(k)}\right\|_2^2$。为了避免过拟合，添加正则项 $L_{\text{reg}} = \sum_{i=1}^{k-1}(\left\|\hat{W}_x^{(i)}\right\|_F^2 + \left\|W_x^{(i)}\right\|_F^2) + \sum_{i=1}^{k-1}(\left\|\hat{W}_y^{(i)}\right\|_F^2 + \left\|W_y^{(i)}\right\|_F^2)$。

最终 NE-EWV3(AutoEncoder) 定义损失函数为 $L = \alpha L_1 + \beta L_2 + \gamma L_3 + L_{\text{reg}}$，$\alpha$、$\beta$、$\gamma$ 为控制损失项权重的超参数。最终节点表示 $\boldsymbol{v} = x_i^{(k)} \oplus y_i^{(k)}$，由 2 个压缩表示拼接得到。

4.5　模型优化

由于 2、3 节中计算损失函数中条件概率函数 p_1、p_2 均使用了 softmax 函数，每次需要对整个数据集进行计算。为了降低计算量，模型采用负采样[1]。目标函数变为如下形式

$$\log\sigma(\boldsymbol{u}^{\mathrm{T}} \cdot \boldsymbol{v}) + \sum_{i=1}^{k} \boldsymbol{E}_{z\sim P(v)}[\log\sigma(-\boldsymbol{u}^{\mathrm{T}} \cdot \boldsymbol{z})] \tag{10}$$

式中：k 表示负样本个数；σ 表示 sigmoid 函数；$P(v) \propto d_v^{3/4}$，d_v 表示节点 v 的出度。梯度算法使用 Adam[19] 对模型进行优化。

5　实验

5.1　实验数据

实验包括了现实网络中的中文、英文数据集。对于中文测试数据，外部词向量使用微信公众号中 800 万篇文章预先训练得到的词向量 (https://spaces.ac.cn/archives/4304)，词表大小 35 万，维度 256 维。英文使用了 Google 发布的在新闻语料中训练得到的词向量 (https://code.google.com/archive/p/word2vec)，词表大小 300 万，维度 300 维。其中，Zhihu 为中文数据集，Cora、HepTh 为英文数据集。

HepTh 数据集：HEP-TH (high energy physics theory) 是 arXiv 发布的公开论文引用网络，随机抽取其中 10 740 篇包含概述的论文，以论文概述作为节点文本信息，以引用关系对节点之间做边。

Zhihu(知乎) 数据集：知乎是国内的问答社区网站，本文使用 CANE[12] 公开的知乎数据集，其中包含 10 000 个爬取的用户作为节点，以用户关注话题的描述文本作为节点文本信息。

数据集统计信息列在表 3，在外部词向量的未登录词统计列在表 4。

表 3　测试数据集统计

Table 3　Dataset statics

数据集	节点个数	边个数	节点标签
HepTh	10 740	16 629	
Zhihu	10 000	43 894	

表 4　数据集未登录词统计

Table 4　OVW statics

数据集	词个数	未登录词个数	未登录词占比/%
HepTh	601 095	222 839	37.1
Zhihu	14 650	6 591	45.0

实验首先在数据集上对链接预测任务进行了实验，并在 Cora 数据集上对节点分类任务进行了实验。

5.2　基线方法

DeepWalk[8] 是 2014 年提出的网络表示学习算法，主要利用随机游走构造节点上下文信息，并利用词向量算法中的 SkipGram 计算网络表示，

Hierarchical Softmax 进行计算优化, DeepWalk 针对网络结构的二阶相似性进行建模。

LINE[9] 利用预定义的概率密度函数对一阶以及二阶相似度进行了建模。为了尽可能体现 LINE 算法的性能, 这里采用 LINE 算法的 1st+2st 的版本, 即包含一阶相似度以及二阶相似度进行建模。

Node2vec[10] 主要针对随机游走过程中的宽度优先以及深度优先做了优化, 通过控制跳转概率参数 p、q 进一步扩展了 DeepWalk 算法。

CANE[12] 算法主要利用互注意力机制以及卷积神经网络对文本进行建模, 学习在不同上下文状态下节点的不同表示。

5.3　测试方法

由于本文模型引入了外部词向量, 为了减少词向量维度变化可能造成的信息损失, 基线模型得到的表示结果维度与词向量维度相同, 本文模型除向量拼接外, 表示维度与词向量维度相同。

对基线方法中的参数设置, 采用 grid search[20] 进行选取。DeepWalk[8] 每个节点开始的随机游走序列为 10, 游走长度 80, skip-gram 窗口为 10。对涉及负采样的方法, 负样本个数设置为 $K=5$。沿用 CANE 中的参数设置, 对 cora、Zhihu 数据集设置 $\alpha=1.0$、$\beta=0.3$、$\gamma=0.3$; HepTh 数据集设置 $\alpha=0.7$、$\beta=0.2$、$\gamma=0.2$, epoch 个数设置为 200。

NE-EWV1 epoch 个数设置为 50。NE-EWV2 中首先采用 TF-IDS 模型计算关键词, 保留关键词个数 15, 对于 Zhihu、Cora 数据集, 设置 $p=1/3$、$q=1/10$, 对于 HepTh 数据集, 设置 $p=1/2$、$q=1/5$, epoch 个数设置为 50。NE-EWV3(AutoEncoder) 损失函数中设置 $\alpha=10$、$\beta=10$、$\gamma=1$, epoch 个数设置为 200。

对链接预测问题, 即根据表示结果还原网络的联通关系, 采用 AUC 作为评价指标[21], AUC 衡量了正确判定正样本与错误判定负样本的概率差异, AUC 指标越大说明模型在二分类问题上表现越好。对节点分类问题, 即根据表示结果对节点分类进行预测, 采用准确率作为评价指标。

5.4　链接预测

在不同的数据集上针对链接预测任务进行了测试, 测试方法是选取一定比例的边和以及这些边中节点的文本信息作为测试数据, 以剩余数据作为测试集。如表 5、6 所示。

表 5　Zhihu 数据集的 AUC 指标 (256 维)

Table 5　AUC values on Zhihu (256 dimensions)

	训练数据百分比/%								
	15	25	35	45	55	65	75	85	95
DeepWalk	0.565 7	0.573 9	0.576 0	0.591 5	0.593 3	0.607 8	0.611 2	0.620 8	0.659 0
LINE	0.514 7	0.536 4	0.564 2	0.628 5	0.644 5	0.651 6	0.678 1	0.693 7	0.706 9
Node2vec	0.533 2	0.541 3	0.571 8	0.609 2	0.629 9	0.687 6	0.697 3	0.694 7	0.729 2
CANE(200epochs)	0.568 3	0.587 7	0.599 8	0.621 3	0.655 8	0.681 3	0.701 4	0.711 8	0.749 7
NE-EWV1	0.657 3	0.667 2	0.663 8	0.674 2	0.678 8	0.680 1	0.691 6	0.710 3	0.721 1
NE-EWV2	0.674 5	0.689 3	**0.698 1**	**0.697 3**	**0.709 3**	**0.746 4**	0.746 0	**0.777 4**	**0.799 0**
NE-EWV3(拼接)	**0.681 7**	0.662 7	0.690 0	0.682 2	0.691 3	0.722 2	0.718 3	0.755 9	0.766 5
NE-EWV3(Autoencoder)	0.680 3	**0.691 5**	0.683 5	0.695 3	0.685 6	0.743 1	**0.747 9**	0.766 8	0.789 3

表 6　HepTh 数据集的 AUC 指标 (300 维)

Table 6　AUC values on HepTh(256 dimensions)

	训练数据百分比/%								
	15	25	35	45	55	65	75	85	95
DeepWalk	0.533 8	0.629 2	0.669 3	0.680 2	0.695 7	0.692 8	0.706 6	0.716 6	0.693 9
LINE	0.503 7	0.509 0	0.510 2	0.529 3	0.530 9	0.535 8	0.565 5	0.608 3	0.627 2
Node2vec	0.660 5	0.685 0	0.695 9	0.703 1	0.706 5	0.697 7	0.693 7	0.685 2	0.710 1
CANE(200epochs)	0.738 0	**0.777 5**	0.790 7	**0.804 3**	**0.820 6**	0.828 3	**0.842 7**	0.852 8	0.853 1
NE-EWV1	0.665 3	0.653 7	0.671 2	0.665 1	0.688 1	0.716 9	0.744 8	0.767 9	0.802 1
NE-EWV2	0.741 5	0.762 4	0.775 9	0.789 2	0.803 9	0.825 1	0.821 7	0.832 7	0.847 9
NE-EWV3(拼接)	**0.763 3**	0.777 2	**0.807 0**	0.785 3	0.812 7	**0.835 4**	0.829 2	**0.853 6**	**0.855 3**
NE-EWV3(Autoencoder)	0.713 6	0.712 0	0.723 9	0.744 9	0.782 5	0.805 5	0.830 8	0.837 8	0.813 7

1) 在中文数据集中本文模型要优于其他基线模型，相比基线算法中 AUC 指标最好的 CANE，AUC 指标提高了 5%~12%。在英文数据集 HepTH 中与性能最好的 CANE 基本相当。

2) 本文使用了在领域无关的通用语料中训练得到的词向量，在 Zhihu 数据集中未登录词占比 45.0%(Zhihu 数据集中包含了话题描述，即包含了大量专有名词)，在 HepTh 数据集中未登录词占比 43.1%。说明本文方法对通用语料有较好适应性，通用文本语料能够提升某些特定领域的文本信息网络表示学习的表示能力。

综上所述，证明了本文模型能够学习到文本信息网络中的有效网络表示，能够有效捕捉网络本身的结构、语义信息，并在不同数据集以及外部词向量上证明了表示的有效性和鲁棒性。

6 结束语

本文提出了基于外部词向量的网络表示模型，将外部词向量引入到文本信息网络的网络表示学习过程中。模型包括 3 个部分：在语义特征空间中学习包含结构特征约束的表示，在结构特征空间学习语义特征约束的表示，以及表示融合部分。本文在现实网络数据集中，以链接预测实验，证明了本文模型可以学习到节点间链接关系的有效表示，而节点间的链接关系也构成了整个网络结构。

在未来的研究工作中，有如下研究方向：未登录词的表示，通用词向量在领域特定任务中往往面临着存在大量未登录词的问题，利用已知词对未登录词进行有效表示，直观上可以提升模型表示容量，从而提升网络表示能力。

参考文献：

[1] CUI Peng, WANG Xiao, PEI Jian, et al. A survey on network embedding[J]. IEEE transactions on knowledge and data engineering, 2019, 31(5): 833–852.

[2] LIBEN NOWELL D, KLEINBERG J. The link prediction problem for social networks[J]. Journal of the American society for information science and technology, 2007, 58(7): 1019–1031.

[3] LANCICHINETTI A, FORTUNATO S. Community detection algorithms: a comparative analysis[J]. Physical review E, 2009, 80: 056117.

[4] BHAGAT S, CORMODE G, MUTHUKRISHNAN S. Node classification in social networks[M]//AGGARWAL C C. Social Network Data Analytics. Boston, MA: Springer, 2011: 115–148.

[5] DONG Xin, HALEVY A, MADHAVAN J, et al. Similarity search for web services[C]//Proceedings of the Thirtieth International Conference on Very Large Data Bases. Toronto, Canada, 2004: 372–383.

[6] BASTIAN M, HEYMANN S, JACOMY M. Gephi: an open source software for exploring and manipulating networks[C]//Proceedings of International AAAI Conference on Weblogs and Social Media. San Jose, California, USA, 2009: 361–362.

[7] PEROZZI B, AL-RFOU R, SKIENA S. Deepwalk: online learning of social representations[C]//Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA, 2014: 701–710.

[8] TANG Jian, QU Meng, WANG Mingzhe, et al. LINE: large-scale information network embedding[C]//Proceedings of the 24th International Conference on World Wide Web. Florence, Italy, 2005: 1067–1077.

[9] GROVER A, LESKOVEC J. node2vec: scalable feature learning for networks[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco, California, USA, 2016: 855–864.

[10] YANG Cheng, LIU Zhiyuan, ZHAO Deli, et al. Network representation learning with rich text information[C]//Proceedings of the 24th International Conference on Artificial Intelligence. Buenos Aires, Argentina, 2015: 2111–2117.

[11] TU Cuchao, LIU Han, LIU Zhiyuan, et al. Cane: context-aware network embedding for relation modeling[C]//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver, Canada, 2007: 1722–1731.

[12] TANG Lei, LIU Huan. Scalable learning of collective behavior based on sparse social dimensions[C]//Proceedings of the 18th ACM Conference on Information and Knowledge Management. Hong Kong, China, 2009: 1107–1116.

[13] BALASUBRAMANIAN M, SCHWARTZ E L, TENENBAUM J B, et al. The isomap algorithm and topological stability[J]. Science, 2002, 295(5552): 7–7.

[14] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality[C]//Proceedings of the 26th International Conference on Neural Information Processing Systems.

Lake Tahoe, Nevada, 2013: 3111–3119.

[15] RECHT B, RÉ C, WRIGHT S J, et al. Hogwild: a lock-free approach to parallelizing stochastic gradient descent[C]//Proceedings of Advances in Neural Information Processing Systems. 2011: 693–701.

[16] CAO Shaosheng, LU Wei, XU Qiongkai. Grarep: learning graph representations with global structural information[C]//Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. Melbourne, Australia, 2005: 891–900.

[17] TU Cunchao, ZHANG Weicheng, LIU Zhiyuan, et al. Max-margin deepwalk: discriminative learning of network representation[C]//Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence. New York, USA, 2006: 3889–3895.

[18] PAGE L, BRIN S, MOTWANI R, et al. The PageRank citation ranking: Bringing order to the web[R]. Palo Alto: Stanford InfoLab, 1999.

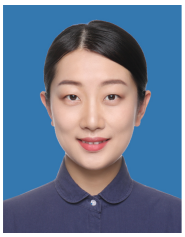
[19] KINGMA D, BA J. Adam: a method for stochastic optimization[C]//Proceedings of the 3rd International Conference on Learning Representations. San Diago, USA, 2015.

[20] HSU C W, CHANG C C, LIN C J. A practical guide to support vector classification. http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf.

[21] HANLEY J A, MCNEIL B J. The meaning and use of the area under a Receiver Operating Characteristic (ROC) curve[J]. Radiology, 1982, 143: 29–36.

作者简介:

张潇鲲, 男, 1991 年生, 硕士研究生, 主要研究方向为网络表示学习。



刘琰, 女, 1979 年生, 副教授, 博士, 主要研究方向为网络信息安全、网络资源测绘。申请发明专利 10 项, 授权 5 项, 发表学术论文 40 余篇。

陈静, 女, 1990 年生, 讲师, 主要研究方向为数据挖掘、自然语言处理和社会网络分析。授权发明专利 2 项, 发表学术论文 10 篇。