



(21) 申请号 202311856289.2

G06F 17/16 (2006.01)

(22) 申请日 2023.12.29

G06N 3/042 (2023.01)

G06N 3/08 (2023.01)

(65) 同一申请的已公布的文献号

申请公布号 CN 117807275 A

(43) 申请公布日 2024.04.02

(73) 专利权人 江南大学

地址 214122 江苏省无锡市滨湖区蠡湖大道1800号

(72) 发明人 詹千熠 王婧 陆恒杨 谢振平
刘渊(74) 专利代理机构 苏州市中南伟业知识产权代
理事务所(普通合伙) 32257
专利代理师 张荣

(51) Int. Cl.

G06F 16/901 (2019.01)

(56) 对比文件

CN 114141375 A, 2022.03.04

CN 113095439 A, 2021.07.09

审查员 周嘉艺

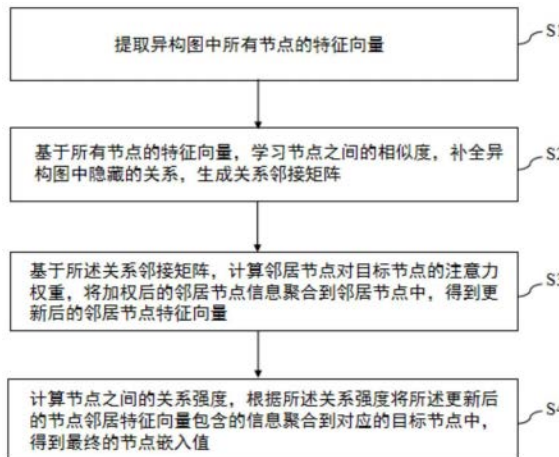
权利要求书5页 说明书12页 附图4页

(54) 发明名称

基于关系挖掘的异构图嵌入方法及系统

(57) 摘要

本发明涉及图数据挖掘技术领域,具体涉及基于关系挖掘的异构图嵌入方法及系统,包括:提取异构图中所有节点的特征向量;基于所有节点的特征向量,学习节点之间的相似度,补全异构图中隐藏的关系,生成关系邻接矩阵;基于所述关系邻接矩阵,计算邻居节点对目标节点的注意力权重,将加权后的邻居节点信息聚合到邻居节点中,得到更新后的邻居节点特征向量;计算节点之间的关系强度,根据所述关系强度将所述更新后的邻居节点特征向量包含的信息聚合到对应的目标节点中,得到最终的节点嵌入值。本发明解决了现有方法过于依赖元路径的先验知识,在节点关系语义聚合时未考虑元路径中间节点的信息的问题,有效利用异构图的拓扑信息,增强了节点之间的联系。



1.一种基于关系挖掘的异构图嵌入方法,其特征在于,包括:

步骤S1:将作者、论文、主题作为不同类型的节点添加到异构图中,添加论文节点和作者节点之间的边以及论文节点和主题节点之间的边,提取异构图中所有节点的特征向量;

步骤S2:基于所有节点的特征向量,学习节点之间的相似度,补全异构图中隐藏的关系,生成关系邻接矩阵;其中,所述关系邻接矩阵的构建方法包括:

步骤S21:将异构图中的所有节点的特征向量 f_v 映射到统一的共享语义空间 f_v' :

$$f_v' = \sigma(f_v \cdot W_{\phi(v)} + b_{\phi(v)})$$

其中, v 表示节点, $\phi(v)$ 表示节点 v 的类型, $W_{\phi(v)} \in R^{d_{\phi(v)} \times d}$ 表示 $\phi(v)$ 类型节点对应的投影矩阵, $b_{\phi(v)}$ 表示偏置参数, $b_{\phi(v)} \in R^{1 \times d}$, $\sigma(\cdot)$ 是非线性激活函数;

步骤S22:对于异构图中的一种关系 $r = \phi(e)$,通过度量学习,学习不同类型节点特征之间的一阶特征相似度 $\Gamma_r^F(f_i', f_j')$;

步骤S23:基于一阶特征相似度 $\Gamma_r^F(f_i', f_j')$,得到相似度图 $S_r^F[i, j]$:

$$S_r^F[i, j] = \begin{cases} \Gamma_r^F(f_i', f_j'), & \Gamma_r^F(f_i', f_j') \geq \varepsilon^F \\ 0, & \text{otherwise} \end{cases}$$

其中, ε^F 表示控制相似度图 $S_r^F[i, j]$ 的稀疏度的参数;

同理计算相同类型的节点之间的特征相似度图:

$$S_r^{FS}[i, j] = \begin{cases} \Gamma_r^{FS}(f_i', f_j'), & \Gamma_r^{FS}(f_i', f_j') \geq \varepsilon^{FS} \\ 0, & \text{otherwise} \end{cases}$$

$$S_r^{FT}[m, n] = \begin{cases} \Gamma_r^{FT}(f_m', f_n'), & \Gamma_r^{FT}(f_m', f_n') \geq \varepsilon^{FT} \\ 0, & \text{otherwise} \end{cases}$$

其中, $S_r^{FS}[i, j]$ 是源节点 s_i 和 s_j 之间的特征相似度图, $S_r^{FT}[m, n]$ 是尾节点 t_m 和 t_n 之间的特征相似度图; $\Gamma_r^{FS}(f_i', f_j')$ 表示源节点 s_i 和 s_j 之间的一阶特征相似度, $\Gamma_r^{FT}(f_m', f_n')$ 表示尾节点 t_m 和 t_n 之间的一阶特征相似度; f_i' 、 f_j' 分别表示源节点 s_i 和 s_j 的特征向量; f_m' 、 f_n' 分别表示尾节点 t_m 和 t_n 的特征向量; ε^{FS} 、 ε^{FT} 分别表示控制特征相似度图 $S_r^{FS}[i, j]$ 、 $S_r^{FT}[m, n]$ 的稀疏度的参数, $\varepsilon^F \in [0, 1]$;

步骤S24:将源节点之间的相似度和尾节点之间的相似度通过初始关系图矩阵的拓扑结构传播,生成传播图:

$$S_r^{FRS} = S_r^{FS} A_r$$

$$S_r^{FRT} = A_r S_r^{FT}$$

步骤S25:采用通道注意力 $w_{\psi, r}^{Feat}$ 对特征相似度图 S_r^F 和传播图 S_r^{FRS} 、 S_r^{FRT} 进行融合,生成关系邻接图 S_r^{Feat} :

$$S_r^{Feat} = \text{soft max}(w_{\psi, r}^{Feat})([S_r^F, S_r^{FRS}, S_r^{FRT}])$$

其中, $w_{\psi, r}^{Feat} \in R^{1 \times 3}$ 为每个关系 r 分配不同的权重;

步骤S26:将关系邻接图 S_r^{Feat} 和同一类型的关系向量 A_r 进行聚合,得到关系邻接矩阵 A

,
r:

$$A_r' = \text{soft max}(w_{\psi,r})([S_r^{\text{Feat}}, A_r])$$

其中, $\text{softmax}(\cdot)$ 表示将权重 $w_{\psi,r}$ 进行归一化;

步骤S3: 基于所述关系邻接矩阵, 计算邻居节点对目标节点的注意力权重, 将加权后的邻居节点信息聚合到邻居节点中, 得到更新后的邻居节点特征向量;

步骤S4: 计算节点之间的关系强度, 根据所述关系强度将所述更新后的邻居节点特征向量包含的信息聚合到对应的目标节点中, 得到最终的节点特征表示, 即节点嵌入值, 利用所述节点嵌入值进行节点分类或链接预测;

其中, 所述节点嵌入值的获取方法包括:

步骤S41: 利用元关系三元组 $\langle \phi(s), \phi(e), \phi(t) \rangle$ 计算节点的关系强度, 得到关系的注意力值 $\text{ATT-head}(s, e, t)$:

$$K(s) = K\text{-Linear}_{\phi(s)}(F[s])$$

$$Q(t) = Q\text{-Linear}_{\phi(t)}(F[t])$$

$$\text{ATT-head}(s, e, t) = (K(s)W_{\phi(e)}^{\text{ATT}}Q(t)^T) \cdot \frac{\mu\langle \phi(s), \phi(e), \phi(t) \rangle}{\sqrt{d}}$$

其中, $\phi(s)$ 表示节点 s 的类型, $\phi(t)$ 表示节点 t 的类型, $\phi(e)$ 表示节点 s 和节点 t 的连接关系类型; $K(s)$ 表示源节点 s 之间的关系强度; $Q(t)$ 表示尾节点 t 之间的关系强度; $F(s)$ 表示源节点 s 的特征表示, $K\text{-Linear}_{\phi(s)}(F[s])$ 表示 $F(s)$ 的线性投影; $F(t)$ 表示尾节点 t 的特征表示, $Q\text{-Linear}_{\phi(t)}(F[t])$ 表示 $F(t)$ 的线性投影; 线性投影 $\text{Linear}_{\phi(v)}$ 用于调整节点 s 或 t 的维度, 将节点维度从 R^d 变成 $R^{\frac{d}{h}}$, h 是注意力头的数量, $\frac{d}{h}$ 是每个头的向量维度;

$\mu\langle \phi(s), \phi(e), \phi(t) \rangle$ 表示调整注意力值的参数;

步骤S42: 将 h 个注意力头连接在一起, 得到每个节点对的注意力向量 $\text{Attention}(s, e, t)$:

$$\text{Attention}(s, e, t) = \text{Soft max}_{\forall s \in N(t)} (\|_{i \in [1, h]} \text{ATT-head}^i(s, e, t))$$

其中, softmax 函数用于对邻居节点 s 的注意力分布进行归一化处理;

步骤S43: 将每个节点对的注意力向量与来自相邻节点的消息向量相结合, 以获得每个目标节点 t 的上下文表示 $\tilde{H}^{(1)}[t]$;

步骤S44: 根据每个目标节点 t 的上下文表示 $\tilde{H}^{(1)}[t]$, 将目标节点 t 映射回其节点类型的分布, 并进行残差连接, 得到更新后的目标节点的上下文表示 $H^{(1)}[t]$, 即最终的节点嵌入值。

2. 根据权利要求1所述的基于关系挖掘的异构图嵌入方法, 其特征在于: 所述节点特征之间的一阶特征相似度 $\Gamma_r^F(f_i', f_j')$ 的获取方法包括:

通过度量学习, 利用节点之间的关系来学习节点特征之间的一阶特征相似度 $\Gamma_r^F(f_i', f_j')$:

$$\Gamma_r^F(f_i', f_j') = \frac{1}{K} \sum_k^K \cos(w_{k,r}^F \otimes f_i', w_{k,r}^F \otimes f_j')$$

其中, f_i' 、 f_j' 均表示经过特征映射之后的节点 v_i 和 v_j 的特征向量; $\otimes$ 代表哈达玛积; $w_{k,r}^F$ 是可学习权重矩阵, 用于加权节点特征; r 表示一种节点之间的关系类型, $r = \varphi(e)$, e 表示节点 v_i 和 v_j 的关系边, $\varphi(\cdot)$ 表示映射函数, $\varphi(e)$ 表示关系边 e 的类型。

3. 根据权利要求1所述的基于关系挖掘的异构图嵌入方法, 其特征在于: 步骤S3中, 得到更新后的邻居节点特征向量的方法为: 设计行归一化矩阵, 根据所述行归一化矩阵计算更新后的节点邻居特征向量 F_r :

$$F_r = \hat{A}_r f_r'$$

其中, $\hat{A}_r$ 表示行归一化矩阵; f_r' 表示在同类型关系 r 中的节点邻居 N_r 的特征向量。

4. 根据权利要求3所述的基于关系挖掘的异构图嵌入方法, 其特征在于: 所述行归一化矩阵 $\hat{A}_r$ 的表达式为:

$$\hat{A}_r = (D_r)^{-1} A_r'$$

其中, A_r' 是关系邻接矩阵, D_r 是 A_r' 的度矩阵, $D_r \in R^{|V_r| \times |V_r|}$ 。

5. 根据权利要求1所述的基于关系挖掘的异构图嵌入方法, 其特征在于: 每个目标节点 t 的上下文表示 $\tilde{H}^{(l)}[t]$:

$$\tilde{H}^{(l)}[t] = \bigoplus_{\forall s \in N(t)} (\text{Attention}(s, e, t) \cdot \text{Message}(s, e, t))$$

其中, $\bigoplus$ 表示连接符号, $\text{Message}(s, e, t)$ 表示来自相邻节点的消息向量。

6. 根据权利要求5所述的基于关系挖掘的异构图嵌入方法, 其特征在于: 所述来自相邻节点的消息向量 $\text{Message}(s, e, t)$:

$$\text{Message}(s, e, t) = \parallel_{i \in [1, h]} \text{MSG-head}^i(s, e, t)$$

其中, h 表示注意力头的数量, $\text{MSG-head}(s, e, t) = M - \text{Linear}_{\phi(s)}(F[s])W_{\varphi(e)}^{\text{MSG}}$ 。

7. 根据权利要求1所述的基于关系挖掘的异构图嵌入方法, 其特征在于: 所述更新后的目标节点的上下文表示 $H^{(l)}[t]$ 为:

$$H^{(l)}[t] = A - \text{Linear}_{\phi(t)}(\sigma(\tilde{H}^{(l)}[t])) + H^{(l-1)}[t]$$

其中, l 表示关系堆叠的层数; $A - \text{Linear}_{\phi(t)}$ 表示线性投影, t 表示目标节点。

8. 一种基于关系挖掘的异构图嵌入系统, 其特征在于, 所述系统用于实现如权利要求1~7任一项所述的基于关系挖掘的异构图嵌入方法, 所述系统包括以下模块:

节点特征向量提取模块, 用于将作者、论文、主题作为不同类型的节点添加到异构图中, 在添加论文节点和作者节点之间的边以及论文节点和主题节点之间的边, 提取异构图中所有节点的特征向量;

关系图生成器, 用于基于所有节点的特征向量, 学习节点之间的相似度, 补全异构图中隐藏的边, 生成关系邻接矩阵; 其中, 所述关系邻接矩阵的构建方法包括:

步骤S21: 将异构图中的所有节点的特征向量 f_v 映射到统一的共享语义空间 f'_v :

$$f'_v = \sigma(f_v \cdot W_{\phi(v)} + b_{\phi(v)})$$

其中, v 表示节点, $\phi(v)$ 表示节点 v 的类型, $W_{\phi(v)} \in R^{d_{\phi(v)} \times d}$ 表示 $\phi(v)$ 类型节点对应的投影矩阵, $b_{\phi(v)}$ 表示偏置参数, $b_{\phi(v)} \in R^{1 \times d}$, $\sigma(\cdot)$ 是非线性激活函数;

步骤S22: 对于异构图中的一种关系 $r = \phi(e)$, 通过度量学习, 学习不同类型节点特征之间的一阶特征相似度 $\Gamma_r^F(f'_i, f'_j)$;

步骤S23: 基于一阶特征相似度 $\Gamma_r^F(f'_i, f'_j)$, 得到相似度图 $S_r^F[i, j]$:

$$S_r^F[i, j] = \begin{cases} \Gamma_r^F(f'_i, f'_j), & \Gamma_r^F(f'_i, f'_j) \geq \epsilon^F \\ 0, & \text{otherwise} \end{cases}$$

其中, ϵ^F 表示控制相似度图 $S_r^F[i, j]$ 的稀疏度的参数;

同理计算相同类型的节点之间的特征相似度图:

$$S_r^{FS}[i, j] = \begin{cases} \Gamma_r^{FS}(f'_i, f'_j), & \Gamma_r^{FS}(f'_i, f'_j) \geq \epsilon^{FS} \\ 0, & \text{otherwise} \end{cases}$$

$$S_r^{FT}[m, n] = \begin{cases} \Gamma_r^{FT}(f'_m, f'_n), & \Gamma_r^{FT}(f'_m, f'_n) \geq \epsilon^{FT} \\ 0, & \text{otherwise} \end{cases}$$

其中, $S_r^{FS}[i, j]$ 是源节点 s_i 和 s_j 之间的特征相似度图, $S_r^{FT}[m, n]$ 是尾节点 t_m 和 t_n 之间的特征相似度图; $\Gamma_r^{FS}(f'_i, f'_j)$ 表示源节点 s_i 和 s_j 之间的一阶特征相似度, $\Gamma_r^{FT}(f'_m, f'_n)$ 表示尾节点 t_m 和 t_n 之间的一阶特征相似度; f'_i, f'_j 分别表示源节点 s_i 和 s_j 的特征向量; f'_m, f'_n 分别表示尾节点 t_m 和 t_n 的特征向量; $\epsilon^{FS}, \epsilon^{FT}$ 分别表示控制特征相似度图 $S_r^{FS}[i, j]$ 、 $S_r^{FT}[m, n]$ 的稀疏度的参数, $\epsilon^F \in [0, 1]$;

步骤S24: 将源节点之间的相似度和尾节点之间的相似度通过初始关系图矩阵的拓扑结构传播, 生成传播图:

$$S_r^{FRS} = S_r^{FS} A_r$$

$$S_r^{FRT} = A_r S_r^{FT}$$

步骤S25: 采用通道注意力 $w_{\psi, r}^{Feat}$ 对特征相似度图 S_r^F 和传播图 S_r^{FRS} 、 S_r^{FRT} 进行融合, 生成关系邻接图 S_r^{Feat} :

$$S_r^{Feat} = \text{soft max}(w_{\psi, r}^{Feat}) ([S_r^F, S_r^{FRS}, S_r^{FRT}])$$

其中, $w_{\psi, r}^{Feat} \in R^{1 \times 1 \times 3}$ 为每个关系 r 分配不同的权重;

步骤S26: 将关系邻接图 S_r^{Feat} 和同一类型的关系向量 A_r 进行聚合, 得到关系邻接矩阵 A_r' :

$$A_r' = \text{soft max}(w_{\psi, r}) ([S_r^{Feat}, A_r])$$

其中, $\text{softmax}(\cdot)$ 表示将权重 $w_{\psi, r}$ 进行归一化;

对象级聚合器, 用于基于所述关系邻接矩阵, 计算邻居节点对目标节点的注意力权重, 将加权后的邻居节点信息聚合到邻居节点中, 得到更新后的邻居节点特征向量;

关系级聚合器, 用于计算节点之间的关系强度, 根据所述关系强度将所述更新后的节

点邻居特征向量包含的信息聚合到对应的目标节点中,得到最终的节点嵌入值,利用所述节点嵌入值进行节点分类或链接预测;其中,所述节点嵌入值的获取方法包括:

步骤S41:利用元关系三元组 $\langle \phi(s), \varphi(e), \phi(t) \rangle$ 计算节点的关系强度,得到关系的注意力值ATT-head(s, e, t):

$$K(s) = K\text{-Linear}_{\varphi(s)}(F[s])$$

$$Q(t) = Q\text{-Linear}_{\varphi(t)}(F[t])$$

$$ATT\text{-head}(s, e, t) = (K(s)W_{\varphi(e)}^{ATT}Q(t)^T) \cdot \frac{\mu\langle \phi(s), \varphi(e), \phi(t) \rangle}{\sqrt{d}}$$

其中, $\varphi(s)$ 表示节点s的类型, $\varphi(t)$ 表示节点t的类型, $\varphi(e)$ 表示节点s和节点t的连接关系类型; $K(s)$ 表示源节点s之间的关系强度; $Q(t)$ 表示尾节点t之间的关系强度; $F(s)$ 表示源节点s的特征表示, $K\text{-Linear}_{\varphi(s)}(F[s])$ 表示 $F(s)$ 的线性投影; $F(t)$ 表示尾节点t的特征表示, $Q\text{-Linear}_{\varphi(t)}(F[t])$ 表示 $F(t)$ 的线性投影; 线性投影 $\text{Linear}_{\varphi(v)}$ 用于调整节点s或t的维度, 将节点维度从 R^d 变成 $R^{\frac{d}{h}}$, h 是注意力头的数量, $\frac{d}{h}$ 是每个头的向量维度;

$\mu\langle \phi(s), \varphi(e), \phi(t) \rangle$ 表示调整注意力值的参数;

步骤S42:将 h 个注意力头连接在一起,得到每个节点对的注意力向量Attention(s, e, t):

$$Attention(s, e, t) = \text{Soft max}_{\forall s \in N(t)}(\|_{i \in [1, h]} ATT\text{-head}^i(s, e, t))$$

其中, softmax函数用于对邻居节点s的注意力分布进行归一化处理;

步骤S43:将每个节点对的注意力向量与来自相邻节点的消息向量相结合,以获得每个目标节点t的上下文表示 $\tilde{H}^{(t)}[t]$;

步骤S44:根据每个目标节点t的上下文表示 $\tilde{H}^{(t)}[t]$,将目标节点t映射回其节点类型的分布,并进行残差连接,得到更新后的目标节点的上下文表示 $H^{(1)}[t]$,即最终的节点嵌入值。

基于关系挖掘的异构图嵌入方法及系统

技术领域

[0001] 本发明涉及图数据挖掘技术领域,尤其是指一种基于关系挖掘的异构图嵌入方法及系统。

背景技术

[0002] 在计算机科学中,图结构是一种基本的数据结构,相对于其他结构更加灵活。图不仅包含个体的属性,还能够捕捉个体之间的相互关系,因此常被用于描述和建模较为复杂的系统。通过多层次分析图结构,用户能够更深入地了解数据背后隐含的信息,从而为各种业务场景的后续任务(如节点分类、链接预测、节点推荐等)提供支持。

[0003] 图嵌入是一种解决图分析问题的有效方法,其将原始图数据转换到低维空间,并保留重要的数据,有助于理解节点之间的关联性,并提高下游任务的性能。

[0004] 为了解决这些挑战,有学者引入了元路径的概念,用于解析不同类型节点之间的语义关系。例如,当一篇论文涉及两位作者时,可以使用元路径 $P=APA$ 来表示。随后,研究者们提出了多种方法。Metapath2vec基于用户定义的元路径进行随机游走,使用异构的Skip-Gram方法学习节点嵌入。MAGNN则手动采样元路径实例,并对节点所在的元路径上的其他节点的语义特征进行融合,将多个元路径的潜在向量聚合到目标节点嵌入中。HAN模型采用将异构图转化为多个子图方式,解决了异构性,提出了分层注意力机制来学习节点嵌入。HHNE模型首次研究双曲空间中HIN嵌入问题,使用元路径引导随机游走来保留HIN结构和语义相关性。MHGCN针对属性复用异构网络进行表示学习,通过多层卷积聚合自动学习HIN中不同长度的元路径,并将多关系结构和属性语义集成到学习的节点嵌入中。考虑到元路径实例之间的依赖关系,MIGTNet对从嵌入目标节点开始的预定义最大元路径长度内的所有元路径进行采样,构建分层图注意力网络,学习节点嵌入。

[0005] 这些模型在异构图嵌入领域取得了一些进展,但仍然存在一些局限性:

[0006] 第一、大多数模型需要特定领域知识来设计异构图的元路径;

[0007] 第二、许多方法依赖一个基本假设,即原始异构图结构是完整的,然而,在数据集预处理过程中,通常会按一定规则对数据进行处理,导致输入模型的图结构可能不是最优的。因此,忽视了数据集的预处理对图的影响。

[0008] 第三、在聚合元路径节点数据时,一些模型仅关注首尾节点,忽略了中间节点携带的语义信息,导致消息传递不全面。

发明内容

[0009] 为解决上述技术问题,本发明提供了一种基于关系挖掘的异构图嵌入方法,所述方法具体步骤包括:

[0010] 步骤S1:提取异构图中所有节点的特征向量;

[0011] 步骤S2:基于所有节点的特征向量,学习节点之间的相似度,补全异构图中隐藏的关系,生成关系邻接矩阵;

[0012] 步骤S3:基于所述关系邻接矩阵,计算邻居节点对目标节点的注意力权重,将加权后的邻居节点信息聚合到邻居节点中,得到更新后的邻居节点特征向量;

[0013] 步骤S4:计算节点之间的关系强度,根据所述关系强度将所述更新后的邻居节点特征向量包含的信息聚合到对应的目标节点中,得到最终的节点嵌入值。

[0014] 在本发明的一个实施例中,步骤S2中,所述关系邻接矩阵的构建方法包括:

[0015] 步骤S21:将异构图中的所有节点的特征向量 f_v 映射到统一的共享语义空间 f'_v :

$$[0016] \quad f'_v = \sigma(f_v \cdot W_{\varphi(v)} + b_{\varphi(v)})$$

[0017] 其中, v 表示节点, $\varphi(v)$ 表示节点 v 的类型, $W_{\varphi} \in R^{d_{\varphi(v)} \times d}$ 表示 $\varphi(v)$ 类型节点对应的投影矩阵, b_{φ} 表示偏置参数, $b_{\varphi} \in R^{1 \times d}$, $\sigma(\cdot)$ 是非线性激活函数;

[0018] 步骤S22:对于异构图中的一种关系 $r = \varphi(e)$,通过度量学习,学习不同类型节点特征之间的一阶特征相似度 $\Gamma_r^F(f'_i, f'_j)$;

[0019] 步骤S23:基于一阶特征相似度 $\Gamma_r^F(f'_i, f'_j)$,得到相似度图 $S_r^F[i, j]$:

$$[0020] \quad S_r^F[i, j] = \begin{cases} \Gamma_r^F(f'_i, f'_j), & \Gamma_r^F(f'_i, f'_j) \geq \varepsilon^F \\ 0, & \text{otherwise} \end{cases}$$

[0021] 其中, ε^F 表示控制相似度图 $S_r^F[i, j]$ 的稀疏度的参数;

[0022] 同理计算相同类型的节点之间的特征相似度图:

$$[0023] \quad S_r^{FS}[i, j] = \begin{cases} \Gamma_r^{FS}(f'_i, f'_j), & \Gamma_r^{FS}(f'_i, f'_j) \geq \varepsilon^{FS} \\ 0, & \text{otherwise} \end{cases}$$

$$[0024] \quad S_r^{FT}[m, n] = \begin{cases} \Gamma_r^{FT}(f'_m, f'_n), & \Gamma_r^{FT}(f'_m, f'_n) \geq \varepsilon^{FT} \\ 0, & \text{otherwise} \end{cases}$$

[0025] 其中, $S_r^{FS}[i, j]$ 是源节点 s_i 和 s_j 之间的特征相似度图, $S_r^{FT}[m, n]$ 是尾节点 t_m 和 t_n 之间的特征相似度图; $\Gamma_r^{FS}(f'_i, f'_j)$ 表示源节点 s_i 和 s_j 之间的一阶特征相似度, $\Gamma_r^{FT}(f'_m, f'_n)$ 表示尾节点 t_m 和 t_n 之间的一阶特征相似度; f'_i, f'_j 分别表示源节点 s_i 和 s_j 的特征向量; f'_m, f'_n 分别表示尾节点 t_m 和 t_n 的特征向量; $\varepsilon^{FS}, \varepsilon^{FT}$ 分别表示控制特征相似度图 $S_r^{FS}[i, j]$ 、 $S_r^{FT}[m, n]$ 的稀疏度的参数, $\varepsilon^F \in [0, 1]$;

[0026] 步骤S24:将源节点之间的相似度和尾节点之间的相似度通过所述初始关系图矩阵的拓扑结构传播,生成传播图:

$$[0027] \quad S_r^{FRS} = S_r^{FS} A_r$$

$$[0028] \quad S_r^{FRT} = A_r S_r^{FT}$$

[0029] 步骤S25:采用通道注意力 $w_{\psi, r}^{Feat}$ 对特征相似度图 S_r^F 和传播图 S_r^{FRS} 、 S_r^{FRT} 进行融合,生成关系邻接图 S_r^{Feat} :

$$[0030] \quad S_r^{Feat} = \text{soft max}(w_{\psi, r}^{Feat})([S_r^F, S_r^{FRS}, S_r^{FRT}])$$

[0031] 其中, $w_{\psi, r}^{Feat} \in R^{1 \times 3}$ 为每个关系 r 分配不同的权重;

[0032] 步骤S26:将关系邻接图 S_r^{Feat} 和同一类型的关系向量 A_r 进行聚合,得到关系邻接矩阵 A'_r :

$$[0033] \quad A'_r = \text{softmax}(w_{\psi,r})([S_r^{Feat}, A_r])$$

[0034] 其中, $\text{softmax}(\cdot)$ 表示将权重 $w_{\psi,r}$ 进行归一化。

[0035] 在本发明的一个实施例中,所述节点特征之间的一阶特征相似度 $\Gamma_r^F(f'_i, f'_j)$ 的获取方法包括:

[0036] 通过度量学习,利用节点之间的关系来学习节点特征之间的一阶特征相似度 $\Gamma_r^F(f'_i, f'_j)$:

$$[0037] \quad \Gamma_r^F(f'_i, f'_j) = \frac{1}{K} \sum_k^K \cos(w_{k,r}^F \otimes f'_i, w_{k,r}^F \otimes f'_j)$$

[0038] 其中, f'_i, f'_j 均表示经过特征映射之后的节点 v_i 和 v_j 的特征向量; $\otimes$ 代表哈达玛积; $w_{k,r}^F$ 是可学习权重矩阵,用于加权节点特征; r 表示一种节点之间的关系类型, $r = \varphi(e)$, e 表示节点 v_i 和 v_j 的关系边, $\varphi(\cdot)$ 表示映射函数, $\varphi(e)$ 表示关系边 e 的类型。

[0039] 在本发明的一个实施例中,步骤S3中,得到更新后的邻居节点特征向量的方法为:设计行归一化矩阵,根据所述行归一化矩阵计算更新后的节点邻居特征向量 F_r :

$$[0040] \quad F_r = \hat{A}_r f'_r$$

[0041] 其中, $\hat{A}_r$ 表示行归一化矩阵; f'_r 表示在同类型关系 r 中的节点邻居 N_r 的特征向量。

[0042] 在本发明的一个实施例中,所述行归一化矩阵 $\hat{A}_r$ 的表达式为:

$$[0043] \quad \hat{A}_r = (D_r)^{-1} A'_r$$

[0044] 其中, D_r 是 A'_r 的度矩阵, $D_r \in R^{|V_r| \times |V_r|}$ 。

[0045] 在本发明的一个实施例中,得到最终的节点嵌入值的步骤包括:

[0046] 步骤S41:利用元关系三元组 $\langle \phi(s), \varphi(e), \phi(t) \rangle$ 计算节点的关系强度,得到关系的注意力值:

$$[0047] \quad K(s) = K\text{-Linear}_{\varphi(s)}(F[s])$$

$$[0048] \quad Q(t) = Q\text{-Linear}_{\varphi(t)}(F[t])$$

$$[0049] \quad ATT\text{-head}(s, e, t) = (K(s) W_{\varphi(e)}^{ATT} Q(t)^T) \cdot \frac{\mu \langle \phi(s), \varphi(e), \phi(t) \rangle}{\sqrt{d}}$$

[0050] 其中, $\varphi(s)$ 表示节点 s 的类型, $\varphi(t)$ 表示节点 t 的类型, $\varphi(e)$ 表示节点 s 和节点 t 的连接关系类型; $K(s)$ 表示源节点 s 之间的关系强度; $Q(t)$ 表示尾节点 t 之间的关系强度; $F(s)$ 表示源节点 s 的特征表示; $F(t)$ 表示尾节点 t 的特征表示;线性投影 $\text{Linear}_{\varphi(v)}$ 用于调整节点 s 或 t 的维度,将节点维度从 R^d 变成 $R^{\frac{d}{h}}$, h 是注意力头的数量, $\frac{d}{h}$ 是每个头的向量维度;

$\mu \langle \phi(s), \varphi(e), \phi(t) \rangle$ 表示调整注意力值的参数;

[0051] 步骤S42:将 h 个注意力头连接在一起,得到每个节点对的注意力向量 Attention

(s, e, t) :

$$[0052] \quad Attention(s, e, t) = \underset{\forall s \in N(t)}{Soft \max}(\|_{i \in [1, h]} ATT-head^i(s, e, t))$$

[0053] 其中, softmax函数用于对邻居节点s的注意力分布进行归一化处理;

[0054] 步骤S43:将每个节点对的注意力向量与来自相邻节点的消息向量相结合,以获得每个目标节点t的上下文表示 $\tilde{H}^{(l)}[t]$;

[0055] 步骤S44:根据每个目标节点t的上下文表示 $\tilde{H}^{(l)}[t]$,将目标节点t映射回其节点类型的分布,并进行残差连接,得到更新后的目标节点的上下文表示 $H^{(l)}[t]$,即最终的节点嵌入值。

[0056] 在本发明的一个实施例中,每个目标节点t的上下文表示 $\tilde{H}^{(l)}[t]$;

$$[0057] \quad \tilde{H}^{(l)}[t] = \bigoplus_{\forall s \in N(t)} (Attention(s, e, t) \cdot Message(s, e, t))$$

[0058] 其中, $\bigoplus$ 表示连接符号, Message(s, e, t)表示来自相邻节点的消息向量。

[0059] 在本发明的一个实施例中,所述来自相邻节点的消息向量Message(s, e, t) :

$$[0060] \quad Message(s, e, t) = \big\|_{i \in [1, h]} MSG-head^i(s, e, t)$$

[0061] 其中, h表示注意力头的数量, $MSG-head(s, e, t) = M - Linear_{\phi(s)}(F[s])W_{\phi(e)}^{MSG}$ 。

[0062] 在本发明的一个实施例中,所述更新后的目标节点的上下文表示 $H^{(l)}[t]$ 为:

$$[0063] \quad H^{(l)}[t] = A - Linear_{\phi(t)}(\sigma(\tilde{H}^{(l)}[t])) + H^{(l-1)}[t]$$

[0064] 其中, l表示关系堆叠的层数; $A - Linear_{\phi(t)}$ 表示线性投影, t表示目标节点。

[0065] 基于同一发明构思,本发明还提供一种基于关系挖掘的异构图嵌入系统,包括以下模块:

[0066] 节点特征向量提取模块,用于提取异构图中所有节点的特征向量;

[0067] 关系图生成器,用于基于所有节点的特征向量,学习节点之间的相似度,补全异构图中隐藏的边,生成关系邻接矩阵;

[0068] 对象级聚合器,用于基于所述关系邻接矩阵,计算邻居节点对目标节点的注意力权重,将加权后的邻居节点信息聚合到邻居节点中,得到更新后的邻居节点特征向量;

[0069] 关系级聚合器,用于计算节点之间的关系强度,根据所述关系强度将所述更新后的节点邻居特征向量包含的信息聚合到对应的目标节点中,得到最终的节点嵌入值。

[0070] 本发明的上述技术方案相比现有技术具有以下优点:

[0071] 本发明能够有效捕获异构图中不同类型节点之间的关系,引入了关系图生成器,有效利用异构图的拓扑信息,并结合对象级聚合和多头注意力机制生成更丰富的节点表示,提出的异构图嵌入方法相比现有技术,具有更好的性能。

附图说明

[0072] 为了使本发明的内容更容易被清楚的理解,下面根据本发明的具体实施例并结合附图,对本发明作进一步详细的说明,其中

- [0073] 图1是本发明实施例中的一种基于关系挖掘的异构图嵌入方法的流程图；
- [0074] 图2是本发明实施例中生成关系邻接矩阵的流程示意图；
- [0075] 图3是本发明实施例中生成所有节点上下文表示向量的流程示意图；
- [0076] 图4是本发明实施例中一种基于关系挖掘的异构图嵌入方法 (HRMNN模型) 的框架流程图；
- [0077] 图5是本发明实施例中链接预测结果图；
- [0078] 图6是本发明实施例中消融实验结果图,其中 (a) 表示在ACM数据集上的实验结果对比, (b) 表示在IMDB数据集上的实验结果对比图, (c) 表示在DBLP数据集上的实验结果对比图。

具体实施方式

[0079] 下面结合附图和具体实施例对本发明作进一步说明,以使本领域的技术人员可以更好地理解本发明并能予以实施,但所举实施例不作为对本发明的限定。

[0080] 实施例一

[0081] 参照图1~3所示,本发明提供了一种基于关系挖掘的异构图嵌入方法,所述方法具体步骤包括:

[0082] 步骤S1:提取异构图中所有节点的特征向量;

[0083] 步骤S2:基于所有节点的特征向量,学习节点之间的相似度,补全异构图中隐藏的关系,生成关系邻接矩阵;

[0084] 步骤S3:基于所述关系邻接矩阵,计算邻居节点对目标节点的注意力权重,将加权后的邻居节点信息聚合到邻居节点中,得到更新后的邻居节点特征向量;

[0085] 步骤S4:计算节点之间的关系强度,根据所述关系强度将所述更新后的邻居节点特征向量包含的信息聚合到对应的目标节点中,得到最终的节点嵌入值。

[0086] 不同类型的节点通常具有各自的特征属性,位于不同的语义空间。直接比较这些不同类型节点的特征可能导致误导性的结果,不能反映节点之间的真实关系。因此,将节点 v 的特征向量 f_v 映射到一个共享的语义空间:

$$[0087] \quad f'_v = \sigma(f_v \cdot W_{\varphi(v)} + b_{\varphi(v)})$$

[0088] 其中, v 表示节点, $\varphi(v)$ 表示节点 v 的类型, $W_{\varphi(v)} \in R^{d_{\varphi(v)} \times d}$ 表示 $\varphi(v)$ 类型节点对应的投影矩阵, $b_{\varphi(v)}$ 表示偏置参数, $b_{\varphi(v)} \in R^{1 \times d}$, $\sigma(\cdot)$ 是非线性激活函数; r 表示一种节点之间的关系类型, $r = \varphi(e)$ 。特征映射操作采用了可学习的参数矩阵,用于处理不同类型节点的特征,以确保各种类型的节点具有一致的表示。这有助于进行节点关系图的分析 and 生成。

[0089] 在异构图中,两个节点特征在特征向量空间中越相似,节点在嵌入空间中彼此距离也越近,属于同一类型的概率越高,这两个相似节点之间存在一条连边的概率也越高。基于异构图的拓扑结构,引入了关系图生成器,利用节点之间的关系来学习节点特征之间的相似度,其原理如图2所示。在图2中仅包含一种关系 $r = \varphi(e)$,该关系涵盖了以下三种关系三元组 $\langle s_i, e_1, t_m \rangle, \langle s_j, e_2, t_m \rangle, \langle s_j, e_3, t_n \rangle$ 。

[0090] 对于关系 r ,通过度量学习,学习不同类型节点特征之间的一阶特征相似度 $\Gamma_r^F(f_i, f_j)$:

$$[0091] \quad \Gamma_r^F(f_i', f_j') = \frac{1}{K} \sum_k^K \cos(w_{k,r}^F \otimes f_i', w_{k,r}^F \otimes f_j')$$

[0092] 其中, f_i', f_j' 均表示经过特征映射之后的节点 v_i 和 v_j 的特征向量; $\otimes$ 代表哈达玛积, $w_{k,r}^F$ 是可学习权重矩阵, 用于加权节点特征; r 表示一种节点之间的关系类型, $r = \varphi(e)$, e 表示节点 v_i 和 v_j 的关系边, $\varphi(\cdot)$ 表示映射函数, $\varphi(e)$ 表示关系边 e 的类型。

[0093] 通过这一过程, 得到了相似度图 S_r^F :

$$[0094] \quad S_r^F[i, j] = \begin{cases} \Gamma_r^F(f_i', f_j'), & \Gamma_r^F(f_i', f_j') \geq \varepsilon^F \\ 0, & \text{otherwise} \end{cases}$$

[0095] 其中, $\varepsilon^F \in [0, 1]$ 控制图的稀疏性。值越接近1, 图就更稀疏, 意味着保留了非常相似的节点。如果 ε^F 较小, 图将包含更多的非零元素, 反映出更多相似的节点。

[0096] 具有相似特征的两个节点可能具有相似的邻居。因此, 针对同一类型的关系 A_r , 用一个元关系三元组表示关系 A_r , 即 $\langle \phi(s), \phi(e), \phi(t) \rangle$, 研究其高阶相似度, 即通过图的结构找到具有相似特征的节点的邻居。

[0097] 因此, 在步骤S2中, 所述关系邻接矩阵的构建方法包括:

[0098] 步骤S21: 计算相同类型的节点 (源结点或尾结点) 之间的相似度, 生成特征相似度图:

$$[0099] \quad S_r^{FS}[i, j] = \begin{cases} \Gamma_r^{FS}(f_i', f_j'), & \Gamma_r^{FS}(f_i', f_j') \geq \varepsilon^{FS} \\ 0, & \text{otherwise} \end{cases}$$

$$[0100] \quad S_r^{FT}[m, n] = \begin{cases} \Gamma_r^{FT}(f_m', f_n'), & \Gamma_r^{FT}(f_m', f_n') \geq \varepsilon^{FT} \\ 0, & \text{otherwise} \end{cases}$$

[0101] 其中, $S_r^{FS}[i, j]$ 是源节点 s_i 和 s_j 之间的特征相似度图, $S_r^{FT}[m, n]$ 是尾节点 t_m 和 t_n 之间的特征相似度图; $\Gamma_r^F(f_i', f_j')$ 表示节点特征之间的一阶特征相似度, $\Gamma_r^{FS}(f_i', f_j')$ 表示源节点 s_i 和 s_j 之间的一阶特征相似度, $\Gamma_r^{FT}(f_m', f_n')$ 表示尾节点 t_m 和 t_n 之间的一阶特征相似度; f_i', f_j' 分别表示源节点 s_i 和 s_j 的特征向量; f_m', f_n' 分别表示尾节点 t_m 和 t_n 的特征向量; $\varepsilon^{FS}, \varepsilon^{FT}$ 均表示控制图的稀疏度的参数, $\varepsilon^F \in [0, 1]$;

[0102] 步骤S22: 将源节点之间的特征相似度图和尾节点之间的特征相似度图通过所述初始关系图矩阵的拓扑结构传播, 生成潜在的传播图:

$$[0103] \quad S_r^{FRS} = S_r^{FS} A_r$$

$$[0104] \quad S_r^{FRT} = A_r S_r^{FT}$$

[0105] 步骤S23: 采用通道注意力 $w_{\psi,r}^{Feat}$ 对特征相似度图 S_r^F 和传播图 S_r^{FRS} 、 S_r^{FRT} 进行融合, 生成关系邻接图 S_r^{Feat} :

$$[0106] \quad S_r^{Feat} = \text{soft max}(w_{\psi,r}^{Feat}) ([S_r^F, S_r^{FRS}, S_r^{FRT}])$$

[0107] 其中, $w_{\psi,r}^{Feat} \in R^{1 \times 1 \times 3}$ 为每个关系 r 分配不同的权重;

[0108] 步骤S24: 为了更全面地考虑关系对图的影响, 进一步提高模型性能, 将关系邻接

图 S_r^{Feat} 和同一类型的关系向量 A_r 进行聚合,得到关系邻接矩阵 A'_r :

$$[0109] \quad A'_r = \text{soft max}(w_{\psi,r})([S_r^{Feat}, A_r])$$

[0110] 其中, $\text{softmax}(\cdot)$ 表示将权重 $w_{\psi,r}$ 进行归一化。关系图生成器通过分析节点之间的连接信息,挖掘特征相似度,并应用通道注意力机制,从而学习到新的关系邻接矩阵。完整的图结构信息有助于更全面地理解节点之间的关系,包括不同节点类型之间的复杂信息。

[0111] 不同类型的邻居节点具有独特的特征和语义,对目标节点的特征产生一定的影响。举例来说,在学术异构图中,一位作者发表的论文与对应的会议之间的关联可以揭示出作者的学术兴趣,进而预测他更倾向于将论文提交到哪些会议。因此,为了确保目标节点(作者节点)信息的完整性,为下游任务提供有力支持,采用对象级聚合模块处理目标节点的邻居节点的信息。

[0112] 在步骤S3中,得到更新后的邻居节点特征向量的方法为:设计行归一化矩阵根据不同邻居节点类型对目标节点的重要性进行加权,所述行归一化矩阵 $\hat{A}_r$ 的表达式为:

$$[0113] \quad \hat{A}_r = (D_r)^{-1} A'_r$$

[0114] 其中, D_r 是 A'_r 的度矩阵, $D_r \in R^{|V_r| \times |V_r|}$ 。

[0115] 根据所述行归一化矩阵 $\hat{A}_r$,针对在相同关系 r 中的节点邻居 N_r ,将其特征向量 f'_r 与相应的自适应权重矩阵相乘,进行聚合,计算更新后的节点邻居特征向量 F_r :

$$[0116] \quad F_r = \hat{A}_r f'_r$$

[0117] 其中, $\hat{A}_r$ 表示行归一化矩阵; f'_r 表示在同类型关系 r 中的节点邻居 N_r 的特征向量。为了符号一致性,更新目标节点的特征,令 $F_t = f'_t$,有助于确保所有的特征都按照相同的方式进行处理。

[0118] 通过行归一化矩阵方法,对象级聚合模块综合考虑了不同邻居节点对目标对象的影响。为后续关系级聚合模块学习不同类型高阶邻居节点的信息,更新目标节点表示提供了基础。

[0119] 在图3中,存在两种不同的关系,分别为 $r1 = \varphi(e1), r2 = \varphi(e2)$,关系三元组 $\langle s, e_1, t \rangle, \langle h, e_2, t \rangle$ 分别与之对应。对于目标对象 t ,来自不同类型邻居对象的信息可能产生不同的影响。因此,为了有效利用不同关系,如图3所示,学习不同类型邻居对象的重要性权重,采用注意力机制聚合来自源节点的信息,以获得目标节点 t 的上下文表示。

[0120] 由此,得到最终的异构图的步骤包括:

[0121] 步骤S41:利用元关系三元组 $\langle \phi(s), \varphi(e), \phi(t) \rangle$ 计算节点的关系强度,得到关系的注意力值:

$$[0122] \quad K(s) = K\text{-Linear}_{\varphi(s)}(F[s])$$

$$[0123] \quad Q(t) = Q\text{-Linear}_{\varphi(t)}(F[t])$$

$$[0124] \quad ATT-head(s, e, t) = (K(s) W_{\varphi(e)}^{ATT} Q(t)^T) \cdot \frac{\mu \langle \phi(s), \varphi(e), \phi(t) \rangle}{\sqrt{d}}$$

[0125] 其中, $\phi(s)$ 表示节点s的类型, $\phi(t)$ 表示节点t的类型, $\phi(e)$ 表示节点s和节点t的连接关系类型; $K(s)$ 表示源节点s之间的关系强度; $Q(t)$ 表示尾节点t之间的关系强度; $F(s)$ 表示源节点s的特征表示; $F(t)$ 表示尾节点t的特征表示; 线性投影 $\text{Linear}_{\phi(v)}$ 用于调整节点s或t的维度, 将节点维度从 R^d 变成 $R^{\frac{d}{h}}$, h 是注意力头的数量, $\frac{d}{h}$ 是每个头的向量维度; 由于异构图中的节点对之间具有多种关系, 因此采用 $W_{\phi(e)}^{ATT} \in R^{\frac{d}{h} \times \frac{d}{h}}$ 作为可学习权重矩阵, 适应不同的边。 $\mu\langle\phi(s), \phi(e), \phi(t)\rangle$ 表示调整注意力值的参数, 以适应不同关系;

[0126] 步骤S42: 为了更全面地捕获不同类型邻居节点的信息, 引入多头注意力机制, 将 h 个注意力头连接在一起, 得到每个节点对的注意力向量 $\text{Attention}(s, e, t)$:

$$[0127] \quad \text{Attention}(s, e, t) = \text{Soft max}_{\forall s \in N(t)} (\|_{i \in [1, h]} \text{ATT-head}^i(s, e, t))$$

[0128] 其中, softmax函数用于对邻居节点s的注意力分布进行归一化处理, 以确保邻居节点s的注意力分布满足归一化, 这便于有效地对不同邻居节点进行加权平均; $\text{ATT-head}^i(s, e, t)$ 表示第 i 个注意力头;

[0129] 步骤S43: 将每个节点对的注意力向量与来自相邻节点的消息向量相结合, 以获得每个目标节点t的上下文表示 $\tilde{H}^{(l)}[t]$:

$$[0130] \quad \tilde{H}^{(l)}[t] = \bigoplus_{\forall s \in N(t)} (\text{Attention}(s, e, t) \cdot \text{Message}(s, e, t))$$

[0131] 其中, $\bigoplus$ 表示连接符号, $\text{Attention}(s, e, t)$ 表示每个节点对的注意力向量, $\text{Message}(s, e, t)$ 表示来自相邻节点的消息向量; 注意力向量将不同类型的邻居节点的信息加权平均, 确保更加关注对目标节点产生最大影响的邻居节点。所述来自相邻节点的消息向量 $\text{Message}(s, e, t)$:

$$[0132] \quad \text{Message}(s, e, t) = \big\|_{i \in [1, h]} \text{MSG-head}^i(s, e, t)$$

[0133] 其中, h 表示注意力头的数量, $\text{MSG-head}(s, e, t) = M - \text{Linear}_{\phi(s)}(F[s])W_{\phi(e)}^{MSG}$ 。

[0134] 步骤S44: 根据每个目标节点t的上下文表示 $\tilde{H}^{(l)}[t]$, 将目标节点t映射回其节点类型的分布, 并进行残差连接, 得到更新后的目标节点的上下文表示 $H^{(l)}[t]$, 即最终的节点嵌入值:

$$[0135] \quad H^{(l)}[t] = A - \text{Linear}_{\phi(t)}(\sigma(\tilde{H}^{(l)}[t])) + H^{(l-1)}[t]$$

[0136] 其中, l 表示关系堆叠的层数; $A - \text{Linear}_{\phi(t)}$ 表示线性投影, t 表示目标节点。

[0137] 下面针对三个典型的异构信息网络数据集设计了多种实验, 并与多个基线模型进行了详尽的比较, 展示本发明方法 (如图4所示的HRMNN模型) 的卓越性能。

[0138] 表1

[0139]	数据集	节点	边
	ACM	paper: 4025 author:17351 subject:72	paper-author:13407 paper-subject:4025
	IMDB	movie:4278 actor:5257 director:2081	movie-actor:12828 movie-director:4278
	DBLP	author:4057 paper:14328 term:7723 conf:20	paper-author:19645 paper-term:85810 paper-conf:14328

[0140] 使用数据集ACM、IMDB、DBLP进行实验,表1总结了三个数据集的具体数据。ACM数据集来自引文网络数据,包含了三种对象类型:作者(A)、论文(P)和主题(S)。论文(P)分为三个研究领域:数据库、数据挖掘和无线通信,用作论文的标签。论文的特征是由其关键词构成的词袋表示,而作者的特征向量则是其连接的论文特征的平均值。主题的特征则使用了one-hot编码。

[0141] IMDB数据集经过预处理,包含了三种对象类型:电影(M)、演员(A)和导演(D)。电影(M)包括三种类型的标签:动作片、喜剧和戏剧。电影的特征由其电影情节关键词构成的词袋表示,而演员和导演的特征是对应的电影特征的平均值。

[0142] DBLP数据集包含了四种对象类型:论文(P)、作者(A)、会议(C)和术语(T)。作者(A)根据其研究领域分为四类:数据库、数据挖掘、人工智能和信息检索。论文的特征是论文标题的词袋表示,作者的特征是从其发表的论文中提取的关键词的词袋表示,术语的特征采用了预训练的单词向量表示,而会议的特征则采用了one-hot向量表示。

[0143] 实验1:节点分类

[0144] 专注于ACM数据集的paper节点、DBLP数据集的作者节点以及IMDB数据集的movie节点,以评估HRMNN模型在节点分类任务中的性能。在本次实验中,采用了不同的训练比例,具体分为20%、40%、60%、80%,将目标节点按照这些比例划分,而其余节点则被用作验证集和测试集。采用了Macro-F1和Micro-F1作为评估指标,实验结果如表2所示。

[0145] 表2

Dataset	Metrics	Train%	ie-HGCN	HGT	HAN	Metapath2vec	GCN	GAT	Node2Vec	HRMNN
[0146] ACM	Macro-F1	20%	0.9165	0.9159	0.9169	0.7265	0.8776	0.8803	0.7492	0.9274
		40%	0.9186	0.9201	0.9230	0.7397	0.8910	0.8802	0.7461	0.9295
		60%	0.9181	0.9183	0.9204	0.7155	0.8721	0.8747	0.7368	0.9228
		80%	0.9350	0.9370	0.9337	0.7504	0.9045	0.9151	0.7615	0.9408
	Micro-F1	20%	0.9161	0.9155	0.9168	0.7350	0.8783	0.8820	0.7397	0.9273

[0147]

		40%	0.9189	0.9205	0.9230	0.7435	0.8924	0.8833	0.7483	0.9288
		60%	0.9182	0.9195	0.9219	0.7354	0.8748	0.8786	0.7407	0.9244
		80%	0.9356	0.9381	0.9356	0.7596	0.9059	0.9158	0.7681	0.9406
IMDB	Macro-F1	20%	0.5991	0.5959	0.5903	0.4657	0.5219	0.5429	0.4948	0.6050
		40%	0.6244	0.6394	0.6467	0.5231	0.5726	0.5996	0.5476	0.6550
		60%	0.6404	0.6767	0.6565	0.5389	0.5637	0.5921	0.5412	0.6877
		80%	0.6775	0.6885	0.6761	0.5663	0.6157	0.6124	0.5724	0.7032
	Micro-F1	20%	0.5991	0.5950	0.5897	0.4683	0.5202	0.5406	0.5018	0.6055
		40%	0.6259	0.6376	0.6493	0.5311	0.5713	0.5963	0.5509	0.6563
		60%	0.6445	0.6756	0.6574	0.5429	0.5618	0.5897	0.5446	0.6864
		80%	0.6761	0.6875	0.6761	0.5674	0.6136	0.6098	0.5897	0.7027
DBLP	Macro-F1	20%	0.8168	0.9048	0.8939	0.7852	0.7763	0.7832	0.7526	0.9096
		40%	0.8364	0.9241	0.9166	0.8175	0.7961	0.7966	0.779	0.9410
		60%	0.8542	0.9304	0.9251	0.8298	0.8041	0.7964	0.781	0.9355
		80%	0.8762	0.9361	0.9386	0.8589	0.8330	0.8368	0.8153	0.9531
	Micro-F1	20%	0.8113	0.9125	0.8954	0.7914	0.7848	0.7934	0.7677	0.9162
		40%	0.8450	0.9323	0.9166	0.8241	0.8032	0.8057	0.7858	0.9459
		60%	0.8608	0.9372	0.9263	0.8344	0.8153	0.8067	0.793	0.9433
		80%	0.8821	0.9410	0.9387	0.8630	0.8428	0.8452	0.8257	0.9582

[0148] 表2的实验结果展示HRMNN模型实现了较好的性能。这证明了通过充分挖掘不同关系对目标节点的影响,能够更有效地学习节点的表示。与此同时,我们也观察到了Metapath2vec模型在ACM和IMDB数据集上的性能相对较差。这强调手动设置元路径的异构网络嵌入方法仍有改进的空间。

[0149] 实验2:节点聚类

[0150] 使用K-Means算法进行聚类。对于ACM数据集,我们将paper节点划分为训练集(808个)、验证集(401个)、和测试集(2816个);对于DBLP数据集,author节点被划分为训练集(400个)、验证集(400个)、和测试集(3257个);对于IMDB数据集,movie节点划分为训练集(400个)、验证集(400个)、和测试集(3478个)。采用了归一化互信息(NMI)和调整兰德系数(ARI)这两个评估指标评估不同算法在节点聚类任务上的性能,实验结果如表3所示。

[0151] 表3

[0152]

Dataset	Metrics	ie-HGCN	HGT	HAN	Metapath2vec	GCN	GAT	Node2Vec	HRMNN
ACM	NMI	0.3698	0.6772	0.6565	0.2556	0.4719	0.6088	0.5299	0.7342
	ARI	0.1849	0.7276	0.7019	0.3108	0.4666	0.6615	0.5973	0.7726
IMDB	NMI	0.0763	0.1188	0.1049	0.0856	0.0604	0.0767	0.0544	0.1227

[0153]

DBLP	ARI	0.0473	0.1289	0.1140	0.0801	0.0595	0.0717	0.0687	0.1290
	NMI	0.1455	0.6433	0.6372	0.6019	0.2994	0.3740	0.6126	0.6595
	ARI	0.0247	0.7128	0.7088	0.6802	0.1835	0.3771	0.6094	0.7280

[0154] 从表3中的结果可以得知,HRMNN模型表现出色,特别是在ACM数据集上,其NMI和ARI分数明显高于其他算法。相反,ie-HGCN表现相对较差,这表明该模型可能不太适用于节点聚类的下游任务。需要注意的是,由于数据集中节点标签的不完整性,所有算法在IMDB数据集上的性能相对较低。值得一提的是,HRMNN在ACM数据集上的表现比HGT算法好。这表明,通过相似性来发现图中隐藏的边,使得HRMNN的方法在节点聚类任务中具有显著的优势。

[0155] 实验3:链接预测

[0156] 当进行链接预测实验时,采用DBLP数据集中的paper节点和author节点之间的边,展示HRMNN模型的性能。我们将已连接的paper-author对视为正节点对,将所有未连接的paper-author对视为负节点对。在训练集中包含了13,751条正节点对,并添加了相同数量

的随机采样的负节点对。验证集和测试集中,正节点对和负节点对的数量也保持一致。在本实验中,使用了两个评估指标,分别是曲线下的面积(AUC)和平均精度(AP),具体结果如图5所示。

[0157] 从图4的实验结果来看,HRMNN在各项性能指标上显著优于所有基线模型。特别地,相较于ie-HCGN和HAN模型,HRMNN模型展现出了更出色的性能,这表明HRMNN具备自动挖掘隐藏关系的能力。Metapath2Vec在异构图情境下表现良好,这暗示元路径采样方法在链接预测的应用中具有一定的优势。HGT、HAN和HRMNN模型均表现出色,这可能是因为这些模型能够利用目标节点周围邻居节点的丰富信息,从而优化节点的表示。

[0158] 实验4:消融实验

[0159] 为了全面评估HRMNN模型的有效性,进行了一系列节点分类的消融实验,深入探讨各组件对性能的影响。因此创建了多个HRMNN模型的变体,包括HRMNN-O模型、HRMNN-S模型、HRMNN-L模型、HRMNN-R模型和HRMNN-M模型,以探究它们在不同方面的性能。具体而言:

[0160] HRMNN-O模型直接使用原始邻接矩阵进行建模,不依赖于关系图生成器。

[0161] HRMNN-S模型在生成新的邻接图时未考虑节点间的差异,将所有权重设置为1。

[0162] HRMNN-L模型采用了一种简单的策略,在对象级聚合中,不采取任何操作,直接进行关系聚合。

[0163] HRMNN-R模型在关系聚合时,将关系映射矩阵权重替换为统一的权重。

[0164] HRMNN-M模型不使用关系级聚合器来生成节点表示。

[0165] 实验结果如图6所示。由图6(a)~(c)可知,HRMNN-O模型在ACM数据集上表现不佳,可能因为ACM数据集经过预处理后,存在单个作者节点,关系图生成器补全了ACM异构网络,从而凸显了该模块在捕捉隐藏节点关系方面的重要性。HRMNN-S模型的性能相对HRMNN较差,强调了在生成新的关系图时应更加关注节点之间的差异性。此外,HRMNN-L的结果表明,采用关系矩阵对邻居节点进行信息聚合是必要的。HRMNN-R模型表现最差,这表明采用多头注意力机制聚合源节点信息时,仍需考虑到边的异构性。HRMNN-M的性能相对HRMNN-R有所提升,但仍不及其他模型,这表明关系级聚合器对提升模型性能具有有效性。

[0166] 实施例二

[0167] 基于与实施例一所述的异构图嵌入方法的同一发明构思,本发明还提供一种基于关系挖掘的异构图嵌入系统,所述系统包括以下模块:

[0168] 节点特征向量提取模块,用于提取异构图中所有节点的特征向量;

[0169] 关系图生成器,用于基于所有节点的特征向量,学习节点之间的相似度,补全异构图中隐藏的边,生成关系邻接矩阵;

[0170] 对象级聚合器,用于基于所述关系邻接矩阵,计算邻居节点对目标节点的注意力权重,将加权后的邻居节点信息聚合到邻居节点中,得到更新后的邻居节点特征向量;

[0171] 关系级聚合器,用于计算节点之间的关系强度,根据所述关系强度将所述更新后的节点邻居特征向量包含的信息聚合到对应的目标节点中,得到最终的节点嵌入值。

[0172] 本领域内的技术人员应明白,本申请的实施例可提供为方法、系统、或计算机程序产品。因此,本申请可采用完全硬件实施例、完全软件实施例、或结合软件和硬件方面的实施例的形式。而且,本申请可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质(包括但不限于磁盘存储器、CD-ROM、光学存储器等)上实施的计算机程序产

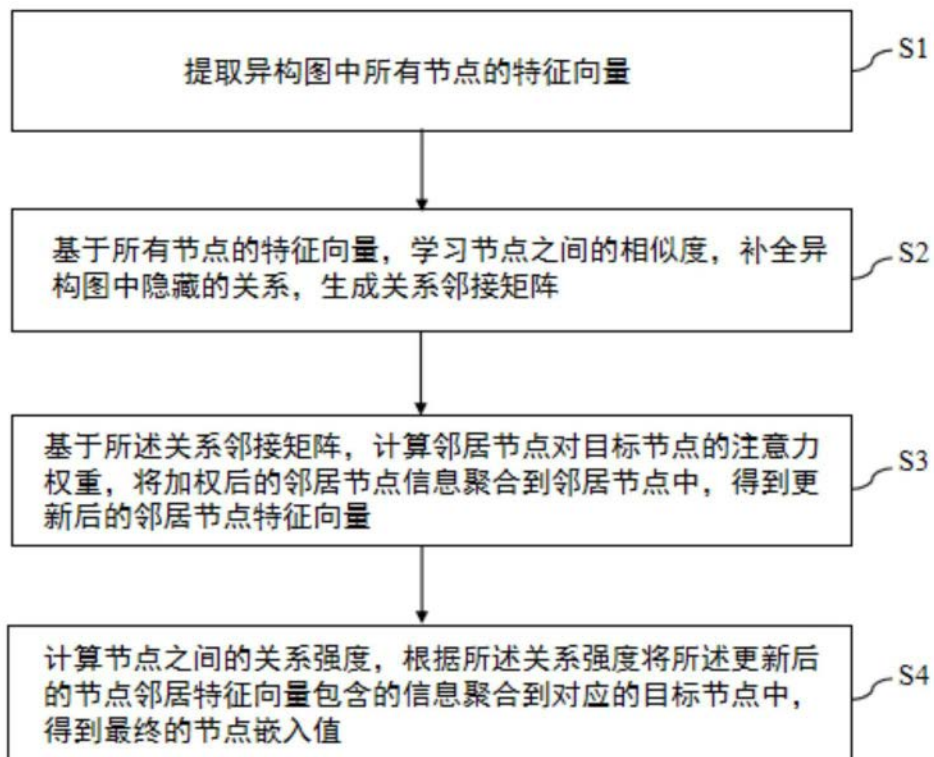
品的形式。

[0173] 本申请是参照根据本申请实施例的方法、设备(系统)、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器,使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[0174] 这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中,使得存储在该计算机可读存储器中的指令产生包括指令装置的制造品,该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[0175] 这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上,使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理,从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[0176] 显然,上述实施例仅仅是为清楚地说明所作的举例,并非对实施方式的限定。对于所属领域的普通技术人员来说,在上述说明的基础上还可以做出其它不同形式变化或变动。这里无需也无法对所有的实施方式予以穷举。而由此所引申出的显而易见的变化或变动仍处于本发明创造的保护范围之内。



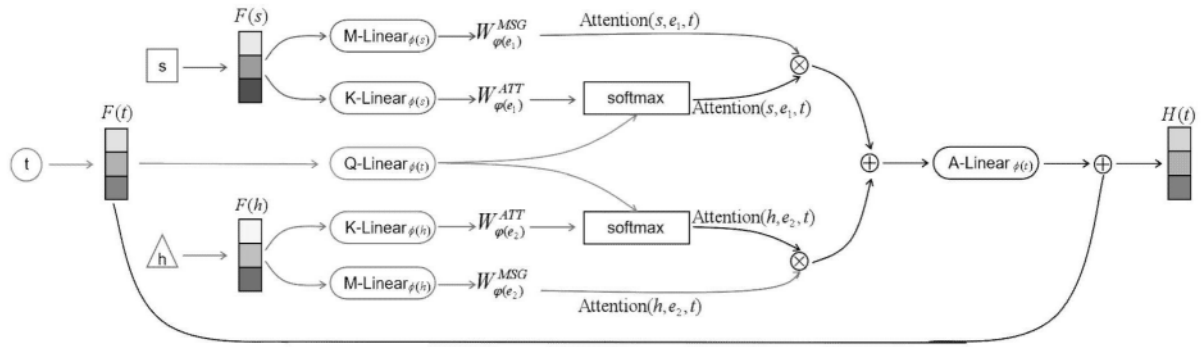


图3

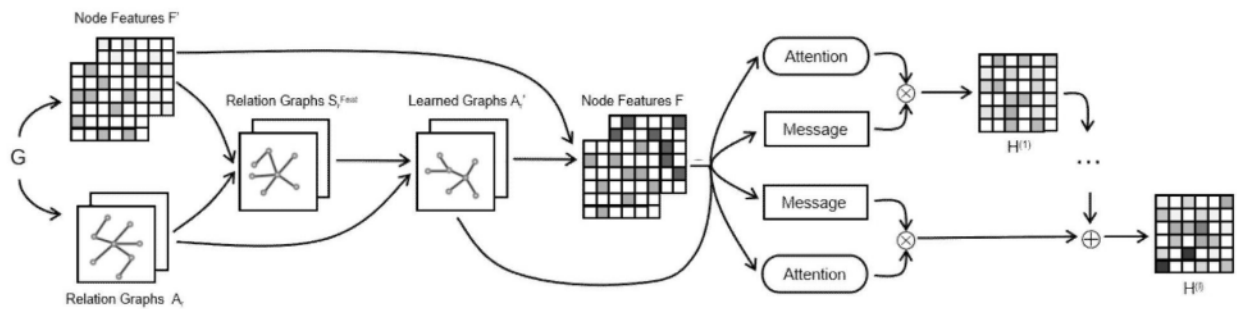


图4

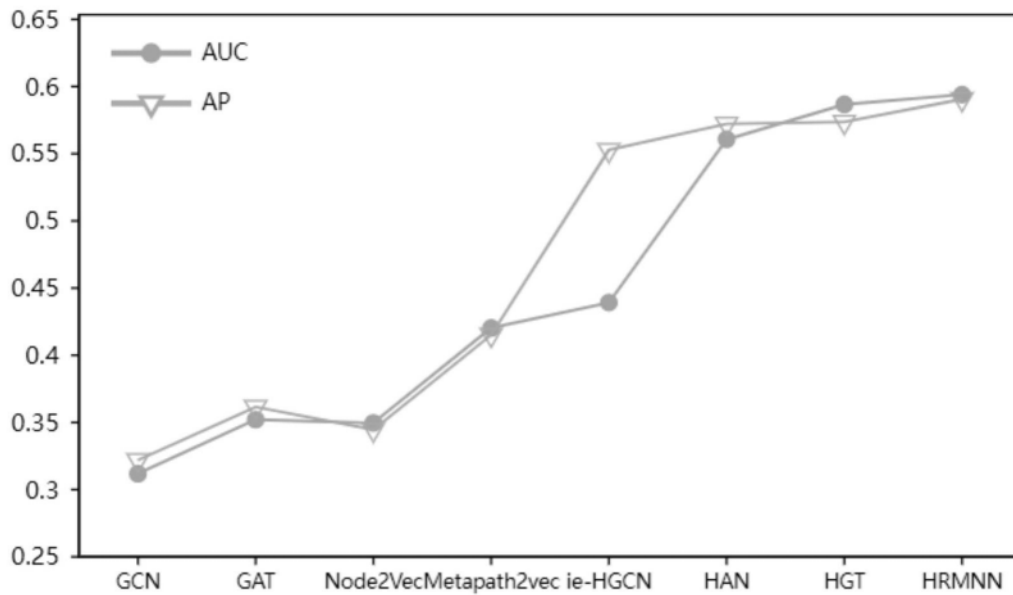
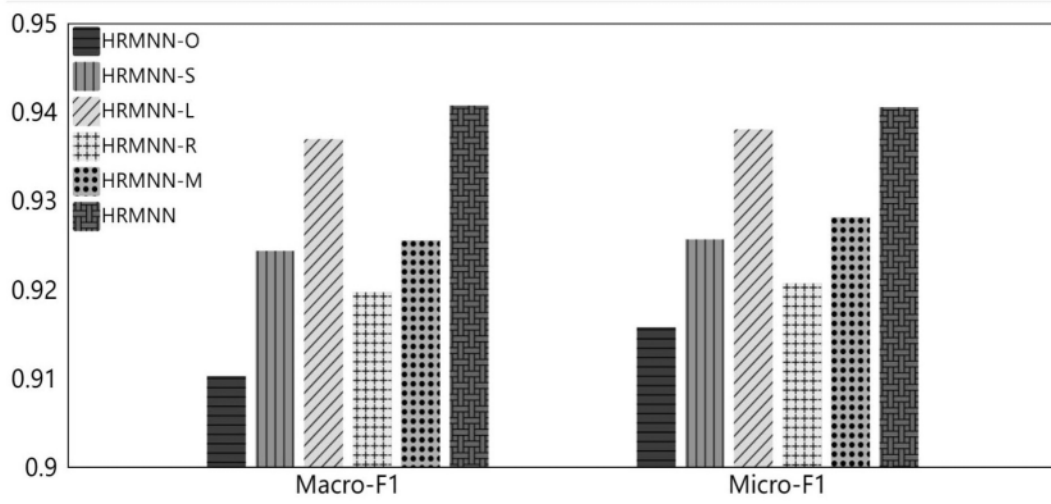
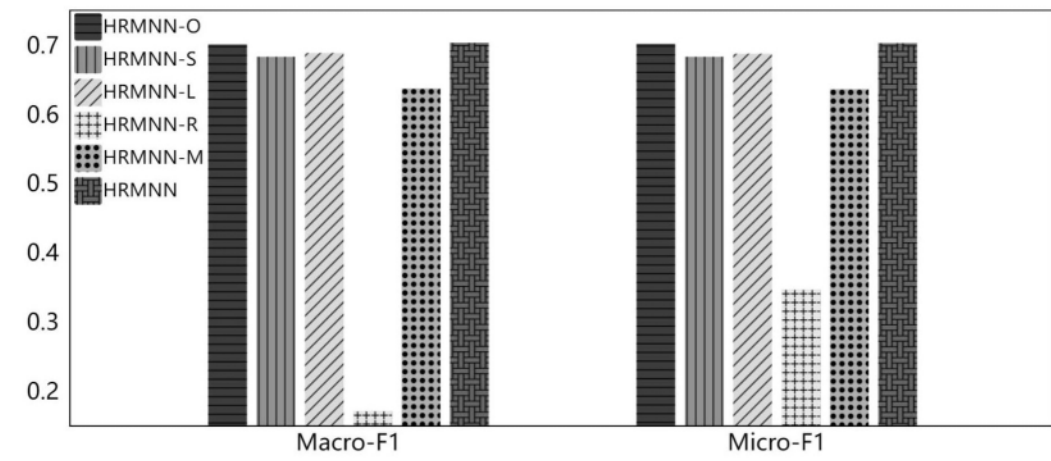


图5



(a)



(b)

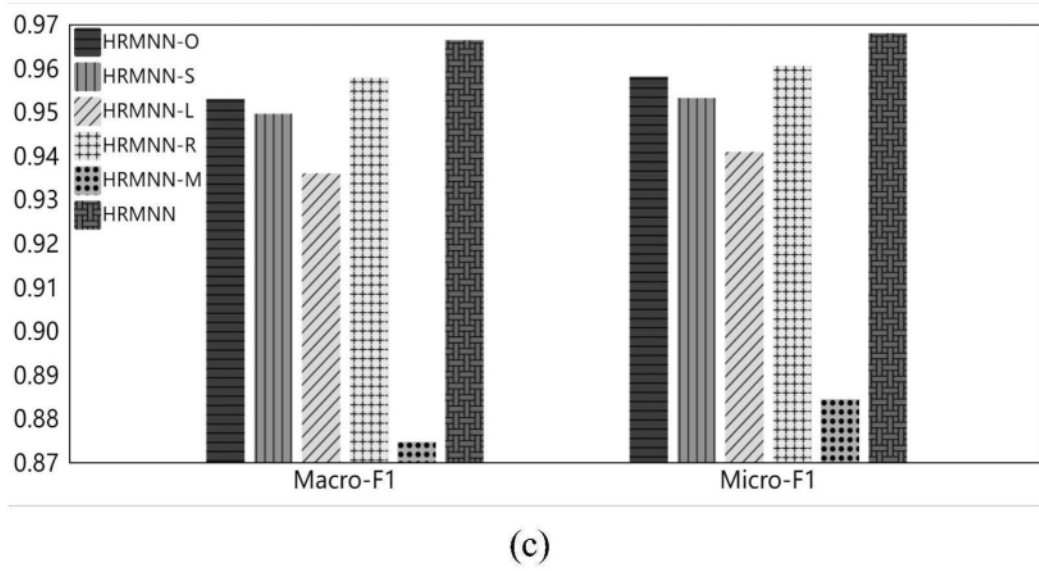


图6