基于图结构与元素信息融合的实体对齐技术

马浩然,王金华*

中国电子科技集团公司第三十二研究所,上海 201808

摘要 实体对齐技术是知识图谱研究中的重要方向,旨在将不同的知识图谱中指向同一现实对象的不同实体进行连接,进而实现知识图谱的扩充。目前,该领域的主流研究思路有2种:一是针对知识图谱的结构特征进行分析,二是针对知识图谱的元素信息(即实体名、关系名、属性名等)进行分析,但是没有同时针对结构特征与元素信息同时进行分析的模型。提出一种实体对齐模型EAFF(entity alignment based on feature fusion),该模型从图结构与元素信息2个角度,对知识图谱进行特征分析。首先,设计了一种基于图神经网络的实体对齐算法,以获得基于图结构的对齐实体对;其次,设计了一种基于元素信息的实体对齐算法,以获得基于元素信息的对齐实体对;最后,利用特征转换与排序算法对2组对齐实体对进行排序,进而得到知识图谱中的对齐实体。在实验中,EAFF取得了相对较好的成绩,优于目前的主流算法。

关键词 知识图谱;实体对齐;图神经网络;特征融合

知识图谱领域的相关研究受到广泛关注,许多机构都建立了不同的知识图谱,如DBpedia、YAGO、ConceptNet、BabelNet和Wikidata。随着大量的知识图谱被建立,如何将这些知识图谱进行融合成为重要问题,实体对齐技术也受到越来越广泛的关注。实体对齐技术是知识融合技术的重要组成部分,旨在将不同的知识图谱中,指向同一对象的不同实体进行连接。在具体的应用场景中,由人工从待对齐图谱中筛选出部分对齐实体作为种子实体对,并利用算法,自动根据种子实体对从知识图谱中发现未被人工标注的对齐实体对。

目前,实体对齐技术有2种主流研究方向。一是利用非语义信息的方法:传统的基于翻译的模型[1]以及基于图神经网络的系列模型[2-3]。二是利

收稿日期:2024-06-30;修回日期:2024-07-26

作者简介:马浩然,硕士研究生,研究方向为知识图谱、自然语言处理,电子信箱:2730173394@qq.com;王金华(通信作者),正高级工程师,研究方向为知识图谱、自然语言处理,电子信箱:15802196002@139.com

引用格式:马浩然, 王金华. 基于图结构与元素信息融合的实体对齐技术[J]. 科技导报, 2024, 42(18): 98-109;
doi:10.3981/j.issn.1000-7857.2024.07.00785

用图谱元素的语义信息,仿照人类的认知逻辑,对图谱中的对齐实体进行学习的方法:基于词嵌入的方法[4-5]以及基于Bert的方法[6-7]。基于知识图谱的结构进行分析的方法极大地受制于知识图谱的异构性;基于图谱元素的名称信息的方法则摒弃了结构信息,这种方法在结构性较强的知识图谱中表现不佳。

通过对知识图谱的结构进行分析发现,种子实体对周围的对齐实体往往同构性与异构性并存[8-10]。图1是取自Wikidata(维基百科)的图谱化数据,种子实体对为"London"——"伦敦"。在中英文图谱中,对齐实体"Great Britain"——"大不列颠"和"United Kingdom"——"英国"与种子实体对都有着相似的结构,存在着同构性;但对齐实体"London University"——"伦敦大学"和"Europe"——"欧洲"与种子实体对的结构存在着较大的不同,存在异构性。如果单纯利用图的结构信息进行实体对齐,对邻居进行编码的过程中,存在异构性的对齐实体会引入较大的噪声;而完全舍弃结构信息,单纯利用元素信息进行分析,则中英文图谱中"London"——"伦敦""Great Britain"——"大不列颠"和"United Kingdom"——"英国"相似的结构特征将被忽略。因此,在进行实体对齐的时候,既要关注到图谱的同构性对于结构分析的促进作用,也要注意到图谱异构性对于元素信息分析的需求,二者缺一不可。

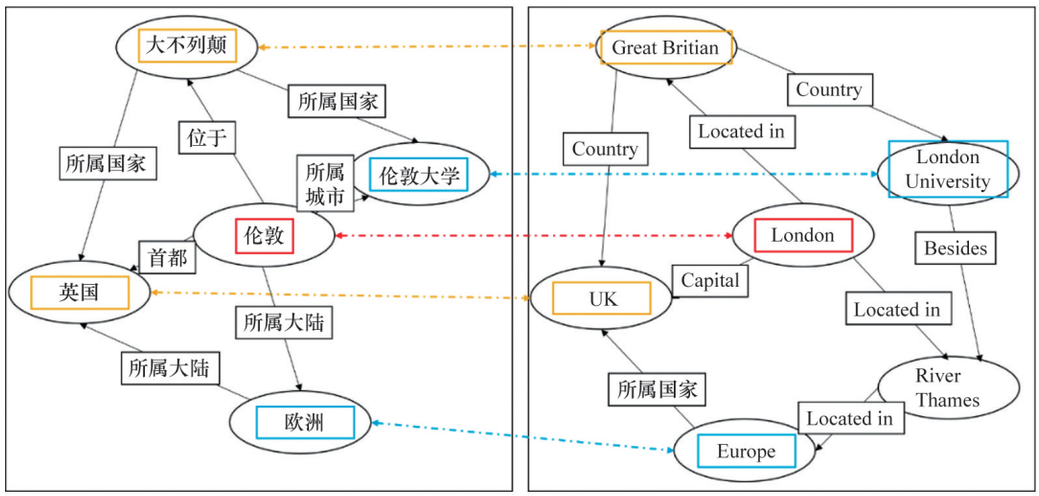

图1　中英文知识图谱实体对齐示例

基于上述特点,本研究设计了一种新的实体对齐模型:首先,为了能更好地学习知识图谱的图结构特征,本模型将图注意力网络(graph attention networks,GAT)与图卷积网络(graph convolutional networks,GCN)进行融合,并利用邻域学习思想,设计了一个新的结构学习算法;其次,为了更好地学习图谱的元素信息特征,设计了一个基于核函数的元素信息学习算法;最后,设计了一种基于特征转换与排序思想的特征融合算法,该算法能融合知识图谱的图结构特征与元素信息特征。

1　研究现状

1.1　基于非语义信息的实体对齐模型

基于非语义信息的实体对齐模型主要有2种类型:翻译模型、图神经网络的结构嵌入模型。

翻译模型中最经典的是TransE模型[11],该模型将关系三元组$\{h,r,t\}$进行嵌入,使得$h+r=t$。在此之后,还有TransH[12]、TransR[13]和TransA[13]等模型,这些工作都是面向TransE[11]在实际工作中出现的各种各样的问题,基于嵌入的思想,进行更新。

基于图神经网络的模型紧跟图神经网络的发展,从GNN(graph neural networks,图神经网络)到GCN再到GAT。EASAE[6]通过累加相邻实体和关系的嵌入来更新实体的嵌入,通过整合头实体和尾实体的嵌入来更新关系的嵌入,在此基础上,RSNs[14]通过生成额外三元组,再利用VR-GCN[15]的方法进行对齐。此外,还有HGCN[16],在利用GCN网络之外,还添加了highway网络,MuGNN[17]在GNN的基础上添加了自注意力机制与图交叉注意力机制。RREA[18]将GNN和翻译模型结合,并将关系嵌入和实体嵌入进行结合来完成实体对齐。NMN[19]在GCN嵌入时,使用跨图注意力机制来获取邻接实体的差异。

1.2 基于语义信息的实体对齐模型

基于语义信息的模型通常情况下,建立于如下假设之上:对齐实体的名称通常相似。因此,异构实体可以通过语义信息进行对齐,以补充结构信息。这一类的模型也可以分为2种:基于普通词嵌入的方法及基于Bert的方法。

基于普通词嵌入的方法有很多,包括:MultiKE[20]基于实体名称、图形结构和实体属性嵌入实体。带自适应特征融合的集体实体对齐框架(CEAFF)[21]融合了3种不同类型的信息,包括结构信息、语义信息和字符串信息,用于表示学习。

基于Bert的方法是目前STOA方法的主要来源,HMAN[22]和BERT-INT[11]都使用多语言Bert来弥合跨语言知识图之间的差距。HMAN[22]采用GCN结合拓扑连接、关系、属性等多方面信息,学习实体嵌入。BERT-INT[11]利用了实体描述的边信息,忽略了知识图中的结构信息。基于语义信息的方法是目前效果最好的一类方法。

目前,在实体对齐领域中,没有综合考虑非语义信息与语义信息的相关研究,缺乏一种能够综合运用知识图谱图结构与元素信息的模型。针对这样的研究现状与现实缺失,进行了相关研究,设计了一种综合图结构与元素信息的实体对齐模型EAFF。

2 问题定义

本研究将知识图谱表示为$G=\{E,R,A,V\}$,E、R、A、V分别表示实体、关系、属性和值的集合。图谱由关系三元组$\{h,r,t\}$和属性三元组$\{h,a,v\}$构成,关系三元组$\{h,r,t\}$中h表示头实体,t表示尾实体,r表示关系,$h,t\in E$,$r\in R$;属性三元组$\{h,a,v\}$中h表示实体,a表示属性,v表示属性值,$h\in E$,$a\in A$,$v\in V$。

给定2个知识图谱$G1$和$G2$,还有一组已经完成对齐的种子实体对$I=\{(e_i,e_i')|e_i\in G1,e_i'\in G2,i\in(0,n)\}$。目的是让模型对种子实体对进行学习,从结构特征以及元素信息2个角度分析出对齐实体的特点,使模型能够针对图谱$G1$中的任意实体,找出对应$G2$中的候选实体。即针对$e_i\in G1$,选出$C=\{(e_{c1}',e_{c2}',\cdots,e_{cn}')|e_{ct}'\in G2,t\in(1,n)\}$。

3 模型架构

3.1 总体架构

如图2所示,模型EAFF可以分为5个部分,分别是:输入模块、结构学习模块、元素信息学习模块、特征融合模块及输出模块。

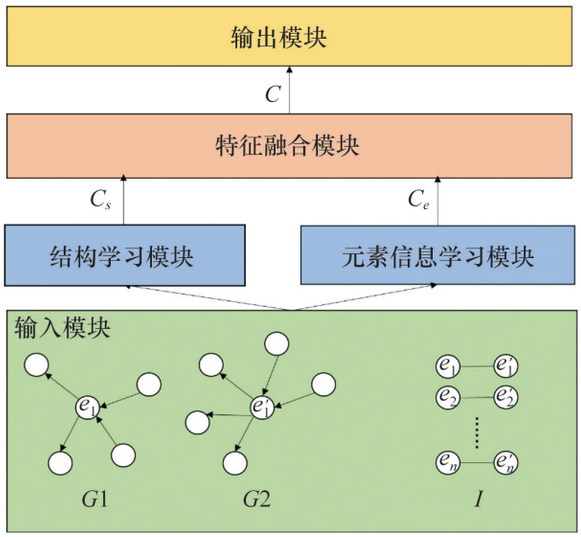

图2 EAFF总体架构图

输入模块负责输入整个模型训练所需的所有数据,包括知识图谱$G1$和$G2$以及对齐的种子实体对$I=\{(e_i,e_i')\}$。输入模块输入的数据有2个去向,其一为结构学习模块,其二为元素信息模块。结构学习模块首先利用GAT+GCN对结构进行学习,而后进行邻域对齐与嵌入重置,实现从结构角度的实体对齐学习。元素信息学习模块,针对种子实体对的实体名、邻居名和属性名进行学习,利用注意力机制,学习到对齐实体的元素名称信息。而后是特征融合模块,该模块利用委员会机制,基于排序的思想,将2种意义不同的特征进行排序。最后的输出模块,输入图谱$G1$或$G2$中的一个实体,给出相对图谱$G2$或$G1$中的候选对齐实体集。

3.2　结构学习模块

结构学习模块是针对知识图谱的结构进行学习的模块,这一模块的设计思想是来源于近期的诸多工作,这些工作都展示了知识图谱的图结构对于实体对齐的促进作用。受到NMN[19]的启发,基于对齐实体的邻居学习,这个模块有4个子部分,分别是结构嵌入模块、筛选模块、邻域嵌入模块和嵌入重置模块,其结构如图3所示。

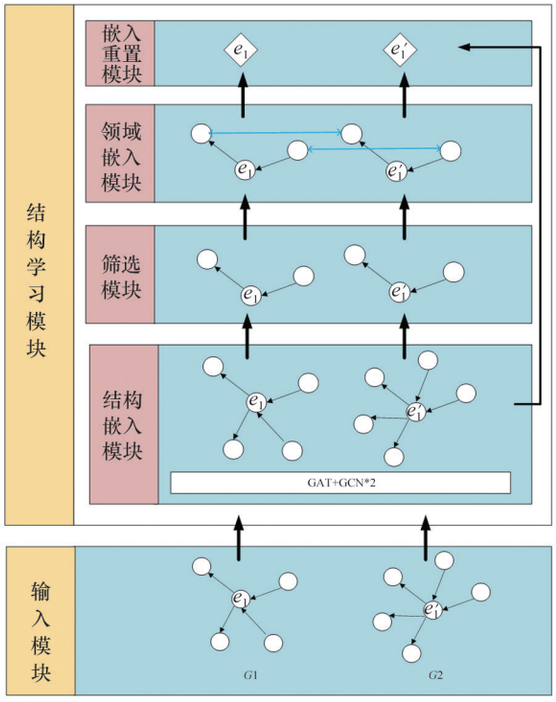

图3　结构学习模块架构图

1)结构嵌入模块。结构嵌入模块的目的是学习对齐实体的结构嵌入,该模块本质上是3层图神经网络结合而成:1层GAT加2层GCN。前者能更好地结合针对某一实体的邻居信息,后者能更好地兼顾知识图谱的全局信息——这对于例如dbp15k这种长尾实体较多的图谱尤为重要。此外,本模块还利用highway网络收束噪声对于结构嵌入的干扰。这一部分的公式如下

$$\alpha_{ij}=\frac{\exp\left(\text{Leaky}ReLU\left(\vec{a}^T\left[\boldsymbol{W}^A\vec{h}_i\,\|\,\boldsymbol{W}^A\vec{h}_j\right]\right)\right)}{\sum_{k\in N_i}\exp\left(\text{Leaky}ReLU\left(\vec{a}^T\left[\boldsymbol{W}^A\vec{h}_i\,\|\,\boldsymbol{W}^A\vec{h}_k\right]\right)\right)} \tag{1}$$

$$\vec{h}_i'=\sigma\left(\sum_{j\in N_i}a_{ij}\boldsymbol{W}^A\vec{h}_j\right) \tag{2}$$

$$\vec{h}_i''=ReLU\left(\sum_{j\in N_i}\frac{1}{\varepsilon}\boldsymbol{W}^{C1}\vec{h}_j'\right) \tag{3}$$

$$\vec{h}_i^*=ReLU\left(\sum_{j\in N_i}\frac{1}{\varepsilon}\boldsymbol{W}^{C2}\vec{h}_j''\right) \tag{4}$$

公式(1)和公式(2)为GAT层的数学表达式,公式(3)和公式(4)为2层GCN层的数学表达式。公式(1)中,$\vec{h}_i$表示最原始的每一个中心实体e_i的特征,$\vec{h}_j$表示中心实体e_i的每一个邻居j的特征,$\boldsymbol{W}^A$表示GAT运算过程中的线性变换矩阵,$\vec{a}^T$表示共享注意力机制,α_{ij}表示中心实体e_i对节点j的注意力参数;公式(2)中,σ是激活函数,$\vec{h}_i'$表示原始数据经过第1层GAT运算后得出的结果;公式(3)中,ε是归一化常数,$\boldsymbol{W}^{C1}$是第1个GCN层的可训练权重矩阵,$\boldsymbol{W}^{C1}\in R^{F''*F'}$,$\vec{h}_i''$表示原始数据经过第1层GAT运算后得出的结果;公式(4)中,$\boldsymbol{W}^{C2}$是第2个GCN层的可训练权重矩阵,$\boldsymbol{W}^{C2}\in R^{F'''*F''}$,$\vec{h}_j''$表示第2层GCN层的输入数据,$\vec{h}_i^*$表示最后的输出数据,也就是中心实体$e_i$经过结构嵌入模块得到的嵌入值。

2)筛选模块。筛选模块分为2部分,分别是邻域筛选模块和候选实体筛选模块,目的是减小对齐实体的邻居数量与候选实体的数量以减小模型的运算量。筛选模块的整体设计思想基于GAT中节点间的注意力机制原理。

邻域筛选模块的设计是基于如下的事实:在知

识图谱中,有一些实体是非常受欢迎的,它的邻居有数十个,如果将这些邻居都进行后续的邻域选择计算,计算量就会非常大。因此,对于知识图谱中的每个实体,都只选择部分邻居作为其一跳邻域,筛选原则基于注意力机制,其数学表达如下

$$a_{i-j}=\frac{\exp\left(\overrightarrow{h_i^*}\boldsymbol{W}_N\overrightarrow{h_j^{*T}}\right)}{\sum_{k\in N_i}\exp\left[\left(\overrightarrow{h_i^*}\boldsymbol{W}_N\overrightarrow{h_k^{*T}}\right)\right]} \tag{5}$$

式中,$\overrightarrow{h_i^*}$表示中心实体e_i经过结构嵌入模块后得到的嵌入,$\overrightarrow{h_j^{*T}}$表示$e_i$的一跳邻居实体$j$经过结构嵌入模块后得到的嵌入的转置,$\boldsymbol{W}_N$是共享权重矩阵。$a_{i-j}$就是邻居$j$对于中心实体$i$的注意力影响值,中心实体$e_i$的所有邻居根据该值排序,根据实际训练中所需的邻居数量n,选出注意力影响值最高的前n个,就是该中心实体e_i经过筛选的一跳邻居,这一部分经过筛选的一跳邻居在下文以m表示。

公式(1)~公式(5)中的N_i表示中心实体e_i的所有邻居实体。

候选实体筛选模块的设计则是基于如下的事实:针对知识图谱$G1$中的实体e_i,如果将其邻域与知识图谱$G2$中的所有实体的邻域进行匹配,计算量会非常大,因此需要筛选出$G2$中和e_i的图嵌入更相关的实体,将这些实体的邻域与e_i的邻域进行后续的匹配。候选实体筛选机制也是基于注意力机制,其数学表达如下

$$a_{i-c}=\frac{\exp\left(\overrightarrow{h_i^*}\boldsymbol{W}_N\overrightarrow{h_c^{*T}}\right)}{\sum_{k\in N_i'}\exp\left[\left(\overrightarrow{h_i^*}\boldsymbol{W}_N\overrightarrow{h_k^{*T}}\right)\right]} \tag{6}$$

式中,对中心实体e_i与候选实体c进行基于注意力的计算,得到α_{i-c},其中$\overrightarrow{h_i^*}$与$\overrightarrow{h_c^*}$分别是中心实体e_i与候选实体c经过结构嵌入模块训练得到的嵌入值。

将候选实体根据α_{i-c}的大小进行排序,排名越靠前,表示候选实体c与中心实体e_i相关度越高。选出排名靠前的前n个候选实体,就是实体经过筛选模块的候选实体集。公式(6)中的N_i'表示知识图谱$G2$中所有的候选实体。

3)邻域嵌入模块。邻域嵌入模块是结构学习模块的核心,该模块的目的通过中心实体e_i的邻域与候选实体e_{ct}'的邻域之间的相互作用,得到中心实体e_i的邻域实体m的邻域对齐嵌入$\overrightarrow{h_m^-}$。这一部分的对齐受到了图匹配和门控神经网络工作的启发。基于之前的工作,重新设计了邻域嵌入的公式,实验结果证明,新的邻域对齐公式能更好地学习实体的邻域特征。该部分的数学表达为

$$a_{m-n}=\frac{exp\left(\overrightarrow{h_m^*}*\overrightarrow{h_n^*}\right)}{\sum_{k\in M_i}exp\left[\left(\overrightarrow{h_m^*}*\overrightarrow{h_k^*}\right)\right]} \tag{7}$$

$$\overrightarrow{h_m^{**}}=\sum_{k\in M_i}a_{m-k}\left[\cos\left(\overrightarrow{h_m^*},\overrightarrow{h_k^*}\right)\overrightarrow{h_k^*}\right] \tag{8}$$

$$\overrightarrow{h_m^-}=\left[\overrightarrow{h_m^*}\middle\|\gamma\times\overrightarrow{h_m^{**}}\right] \tag{9}$$

$$\overrightarrow{h_i^\#}=\boldsymbol{W}_A\left[\sum_{k\in N_i''}\sigma\left(\overrightarrow{h_k^-}W_G\right)\times\overrightarrow{h_m^-}\right] \tag{10}$$

上述公式中的m指知识图谱$G1$中实体e_i经过邻域筛选模块训练后的一个一跳邻居实体,n表示知识图谱$G2$中针对实体e_i的候选实体e_{ct}'经过邻域筛选模块训练后的一个一跳邻居实体。$\overrightarrow{h_m^*}$与$\overrightarrow{h_n^*}$分别指m、n经过结构嵌入模块训练的嵌入值,a_{m-n}指注意力权重。$\overrightarrow{h_m^{**}}$表示实体$m$的邻域矫正嵌入,$\gamma$是根据模型训练情况自定义的邻域矫正向量的参数,将结构嵌入模块学习到的原始的g嵌入$\overrightarrow{h_i^*}$与邻域矫正嵌入$\gamma\times\overrightarrow{h_m^{**}}$进行拼接,得到$e_i$的邻居$m$的邻域对齐嵌入$\overrightarrow{h_m^-}$。利用门控神经网络将$e_i$所有邻居$m$的邻域对齐嵌入$\overrightarrow{h_m^-}$聚合,得到$e_i$的邻域对齐嵌入$\overrightarrow{h_i^\#}$,$\boldsymbol{W}_G$是门控神经网络矩阵,$\boldsymbol{W}_A$是嵌入重置模块的神经网络参数矩阵,$\overrightarrow{h_i^\#}$是实体$e_i$根据邻域对齐嵌入更新的重置嵌入。公式(7)~公式(8)中的M_i表示知识图谱$G2$中针对中心实体e_i经过邻域筛选模块训练后的候选实体e_{ct}'的邻居,公式(10)中的N_i''表示知识图谱$G1$经过筛选模块得到的所有的邻域实体m。

4)嵌入重置模块。最后是嵌入重置模块,利用门控神经网络,将实体e_i的所有邻居m的邻域对齐嵌入$\overrightarrow{h_m^-}$进行聚合,将之作为e_i的邻域嵌入。并将实体e_i的邻域嵌入与结构嵌入模块得到的图嵌入进行结合,这一部分的数学表示为

$$\vec{v}_l = \left[\vec{h}_i^{\#} \middle\| \vec{h}_i^{*}\right] \tag{11}$$

最后将邻域聚合的重置嵌入$\vec{h}_i^{\#}$与结构嵌入模块学习到的原始的嵌入$\vec{h}_i^{*}$进行拼接,得到结构学习嵌入$\vec{v}_l$。

根据结构学习嵌入,就可以进行最后的候选实体选择,其数学公式如下

$$dis = d\left(\vec{v}_i, \vec{v}_c\right) \tag{12}$$

式中,$\vec{v}_i$表示图谱$G1$中的实体i的结构学习嵌入,$\vec{v}_c$表示图谱$G2$中对齐候选实体c的结构学习嵌入,$d\left(\vec{v}_i, \vec{v}_c\right)$表示对实体$i$与实体$c$的结构学习嵌入进行曼哈顿距离计算,$dis$就是最后的曼哈顿距离,该数值越小,表示两个实体之间越相似,也越有可能对齐。

3.3　元素信息学习模块

元素信息学习模块针对知识图谱的元素信息:实体名、邻居名、关系名和属性名进行学习。这一部分所使用的元素信息是基于Bert训练得到的词向量。元素信息模块旨在让图谱$G1$和$G2$中的实体e、e'的元素信息进行融合,以获得实体之间的元素信息的特征,进而确定二者是否为对齐实体。元素信息学习模块分为4部分,分别是实体名融合模块、邻居名融合模块、关系名融合模块和属性名融合模块。模块架构如图4所示。

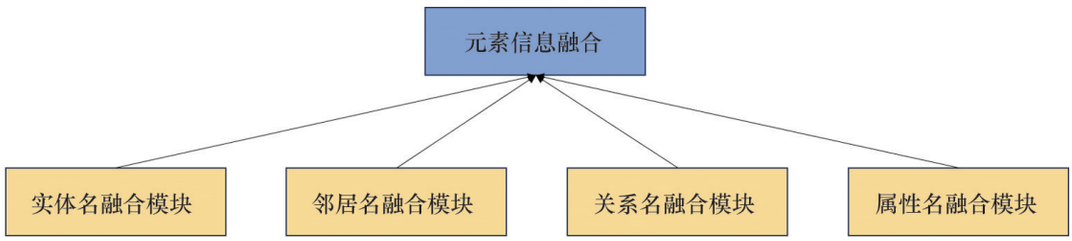

图4　元素信息学习模块架构图

1)实体名融合模块。实体名融合模块就是将待对齐实体e,e'的实体名词向量进行融合,其数学表达如下

$$D(e, e') = \cos(e, e') \tag{13}$$

式中,$\cos(e,e')$表示对e、e'的词向量进行余弦相似度的计算,$D(e,e')$表示待对齐实体e、e'的实体名的相似度。

2)邻居名融合模块。邻居名融合模块是针对对齐实体e、e'的邻居的实体的名称进行融合,这部分的研究参考了BERT-INT[11]的相关工作,其流程如图5所示。

定义实体e的邻居集合为$N[e]=\{e_0,e_1,\cdots,e_n|e_i\in G1\}$,实体$e'$的邻居集合为$N[e']=\{e_0',e_1',\cdots,e_m'|e_j'\in G2\}$,对$N[e]$与$N[e']$中的所有元素进行相似度计算,得到邻居名聚合矩阵$\boldsymbol{C}$,其中的元素表示为$c_{ij}$,其数学表达式为

$$c_{ij} = \cos\left[N\left(e_i\right), N\left(e_j'\right)\right] \tag{14}$$

矩阵$\boldsymbol{C}$的具体细节如图6所示,$N(e_i)$表示$N[e]$中的元素,$N(e_j')$表示$N[e']$中的元素,c_{ij}表示对齐实体e、e'的邻居之间进行余弦相似度计算的值。由此,会得到一个邻居聚集矩阵$\boldsymbol{C}$,$\boldsymbol{C}$是一个$k\times k$的矩

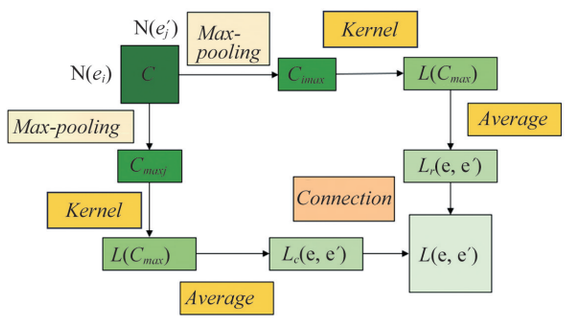

图5　邻居名融合计算流程图

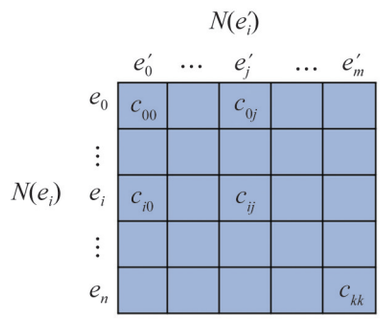

图6 矩阵$\boldsymbol{C}$具体结构

阵,k=max(n,m),没有的元素置0补位。而后,基于矩阵$\boldsymbol{C}$进行双向融合操作,也就是从行和列2个维度对矩阵进行融合。行和列的融合操作一致,行融合的公式为

$$c_{i\max}=\max\left(c_{i0},c_{i1},\cdots,c_{ij},\cdots,c_{ik}\right) \tag{15}$$

$$K_p\left(c_{i\max}\right)=\exp\left[-\frac{\left(c_{i\max}-\mu_p\right)^2}{\sigma_p}\right] \tag{16}$$

$$L\left(\boldsymbol{C}_{\max}\right)=\left[K_0\left(c_{i\max}\right),K_1\left(c_{i\max}\right),\cdots,K_q\left(c_{i\max}\right)\right] \tag{17}$$

$$L_r\left(e,e'\right)=\frac{1}{k}\sum_{i=1}^{k}\log L\left(\boldsymbol{C}_{\max}\right) \tag{18}$$

公式(15)首先对邻居聚集矩阵$\boldsymbol{C}$的每一行都进行最大池化操作,得到最大相似值$c_{i\max}$;最大池化操作可以减少运算量,提高运算速度,并且提高特征的鲁棒性。公式(16)借用核函数,将池化值$c_{i\max}$进行拉普拉斯核函数的转换,在这一步,通过调整参数μ_p、σ_p,设计多个拉普拉斯核(下角标p的不同表示不同的拉普拉斯核),得到多个映射特征,也就是$K_p(c_{i\max})$;将特征从低维空间映射至高维,以便后续的特征运算。公式(17)将每一行的最大相似值$c_{i\max}$经过多个拉普拉斯核运算得到的结果$K_p(c_{i\max})$进行集合,得到$L(\boldsymbol{C}_{\max})$。公式(18)从列方向,将每一行的向量特征进行平均,得到$L_r(e,e')$,即行融合特征。列方向的操作流程与行方向一致,为$L_c(e,e')$。行融合特征与列融合特征的数学表示为

$$L\left(e,e'\right)=L_r\left(e,e'\right)\oplus L_c\left(e,e'\right) \tag{19}$$

式中,$L(e,e')$表示对齐实体e,e'的邻居名融合结果。

3)关系名融合模块与属性名融合模块。关系名融合模块与属性名融合模块的计算流程与思路与邻居名融合模块的操作一致,是将对齐实体e、e'的关系名与属性名分别进行融合,因为篇幅限制,故此不列出具体的数学表达。r_{ij},a_{ij}表示e、e'的关系聚集矩阵$\boldsymbol{R}$和属性聚集矩阵$\boldsymbol{A}$中的元素,其余操作与邻居聚集矩阵的操作一致,最后得到对齐实体e、e'的关系名融合结果$G(e,e')$和属性名融合结果$S(e,e')$。

在得到对于e,e'的实体名融合结果$D(e,e')$、邻居名融合结果$R(e,e')$、关系名融合结果$G(e,e')$和属性名融合结果$S(e,e')$之后,将四者拼接,得到对齐实体e,e'的总相似度$T(e,e')$,其数学表达为

$$T\left(e,e'\right)=\left[D\left(e,e'\right)\oplus R\left(e,e'\right)\oplus G\left(e,e'\right)\oplus S\left(e,e'\right)\right] \tag{20}$$

式中,$T(e,e')$的数值表示实体e、e'的对齐概率,数值越高则对齐的概率越大。

3.4 特征融合模块

针对知识图谱$G1$中的实体e_i,当该实体经过结构学习模块和元素信息嵌入模块后,会分别得到2组候选实体。由于2组候选实体的评价指标(也就是对齐概率)的计算逻辑不同,因此,不能对2组候选实体之间的相似度直接进行计算。基于这个情况,设计了一个排序算法,其设计思路如下。

模型的评价指标是Hits@1和Hits@10,因此,针对每个实体e,结构学习模块和元素信息嵌入模块会分别选出10个候选实体,即结构学习模块候选实体集C_s={c_{s0},c_{s1},…,c_{s9}}和元素信息嵌入模块候选实体集C_e={c_{e0},c_{e1},…,c_{e9}}。将2组候选实体按照各自的评价指标进行排序,然后用序号0~9对候选实体概率由大至小标号,表示如下

$$C_s=\left\{\left(c_{s0},0\right),\left(c_{s0},1\right),\cdots,\left(c_{s9},9\right)\right\} \tag{21}$$

$$C_e=\left\{\left(c_{e0},0\right),\left(c_{e0},1\right),\cdots,\left(c_{e9},9\right)\right\} \tag{22}$$

由此,C_s和C_e两组基于不同逻辑排序的候选实体集就有了共同的比较指标,基于上述逻辑,设计排序算法如下(以Python作为伪代码语言)。

算法一:针对特征融合的排序算法

Input: C_s={(c_{s0},0),(c_{s1},1),…,(c_{s9},9)}

C_e={(c_{s0},0),(c_{s1},1),…,(c_{s9},9)}

Output: 总候选实体集C

(1) for i in range (0,9):

（2） for j in range (0,9):
（3） if c_{si}==c_{ej} :
（4） C.append ((c_{si}, i+j))
（5） elif c_{si}!=c_{ej} and j == 9
（6） C.append ((c_{si}, i+10))
（7） elif i==9
（8） C.append ((c_{ej}, j+10))
（9） for k in range(len(C)):
（10） for l in range(len(C)−k−1)
（11） if l>(l+1)
（12） c_l, $c_{(l+1)}$=$c_{(l+1)}$,c_l

其中,如果c_{si}或c_{ej}能与C_e或者C_s中的某一项指向相同实体,则将二者后面的标号相加;算法的(5)(6)(7)(8)表示:如果c_{si}或c_{ej}在C_e或者C_s中并没有对应指向相同实体的某一项,则将c_{si}或c_{ej}后的序号加10。此时会得到总的候选实体集C

$$C=\left\{\left(c_{l0},l0\right),\left(c_{l1},l1\right),\cdots,\left(c_{lo},lo\right)\mid o\leqslant 20\right\} \tag{23}$$

此时候选实体集C中元素的前后顺序并没有任何的大小关系,因此公式(9)~公式(12)设计了冒泡排序。完成此步骤后,候选实体集C的顺序就是综合评定C_s和C_e中所有候选实体与实体e的对齐概率,该排序由大到小,越靠前的,越可能对齐。

值得注意的是,结构学习模块和元素信息模块的损失函数是一致的,其数学表达为

$$L=\sum_{\left(e,e'^{-},e'^{+}\right)}\max\left\{0,l\left(e,e'^{+}\right)-l\left(e,e'^{-}\right)+\rho\right\} \tag{24}$$

$$l\left(e,e'\right)=\left\|e-e'\right\|_{L1} \tag{25}$$

式中,损失函数主要考虑正确对齐的实体对以及错误对齐实体对于整体的影响,$l(e,e')$表示e与e'的L_1距离。需要注意,在结构学习模块和元素信息模块中,e与e'的嵌入表示不同。

4 实验设计与结果分析

4.1 实验设置与参数确定

本实验使用的语料是跨语言知识库DBP15k,该语料可分为3组,分别是ZH-EN、JA-EN、FR-EN。使用该语料可以更好的对比本模型与其他模型的效果,表1列出了DBP15k的各项数据量。

表1 DBP15k语料信息

语言名称	语料名称					
	ZH-EN		JA-EN		FR-EN	
	ZH	EN	JA	EN	FR	EN
ENT	66469	98125	65744	95680	66858	105889
REL	2830	2317	2043	2096	1379	2209
TRI	153929	237674	164373	233319	192191	278590

本实验选择Hits@1和Hits@10作为评价模型效果的指标,Hits@k的计算如下:测试集中有n个对齐实体对$E=\{(e_i,e_i')\mid i\in(1,n)\}$,将$e_i$放入训练好的模型中进行训练,会得到10个候选实体$C_i$,$C_i=\{c_{i0},c_{i1},\cdots\cdots,c_{i9}\}$(对于$c_{ij}$与$c_{ik}$,如果$j<k$,则$c_{ij}$与$e_i$对齐的概率更大)。对于$n$个实体$e_i$,有$n$组候选实体$C_i$,假设存在$p$组$C_i$,使$c_{i0}$与$e_i'$一致,存在$q$组$C_i$,存在元素与$e_i'$一致,则有

$$\text{Hits@1}=\frac{p}{n} \tag{26}$$

$$\text{Hits@10}=\frac{q}{n} \tag{27}$$

参数设置如下:结构学习模块中,γ=0.1,邻域聚合的重置嵌入$\overrightarrow{h_l^{\#}}$维数为25;元素信息学习模块中,每一组拉普拉斯核都有20个,μ从0.05到0.95,区间跨度为0.05,σ为0.1。因为本实验的评价指标为Hits@1和Hits@10,因此,为了加速实验的计算,结构学习模块和元素信息学习模块在计算过程中都只选择10个候选实体,最后的损失函数ρ设置为1.0。

本实验使用的硬件设施为:Intel Core i7-11800H、RTX3060。此外还使用了Google Colab辅助计算。本实验的语言基于Python,在Windows环境下运行。

4.2 基线模型对比实验

为了验证本模型EAFF相较于之前的工作的提升效果,实验结果如表2所示。

上述基线模型中,TransEdge[23]、MuGNN[17]、KECG[24]、AliNet[25]和NMN[19]属于基于图结构的模型,DGMC[26]、MultiKE[20]、JarKA[27]和SelfMEA[28]属于基于元素信息的模型,GEEA[29]、UMAEA[30]、MEAformer[31]、DESAlign[32]和LLMEA[33]为多模态或大模型实体对齐模型,EAFF为本研究的模型。

表2 各基线模型效用对比

语料名称	准确率									平均值
	DBP_{ZH-EN}			DBP_{JA-EN}			DBP_{FR-EN}			
	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	
TransEdge	73.5	91.9	80.1	71.9	93.2	79.5	71.0	94.1	79.6	81.6
MuGNN	49.4	84.4	61.1	50.1	85.7	62.1	49.5	87.0	62.1	65.7
KECG	47.8	83.5	59.8	49.0	84.4	61.0	48.6	85.1	61.0	64.4
AliNet	53.9	82.6	62.8	54.9	83.1	64.5	55.2	85.2	65.7	67.5
NMN	73.3	86.9	—	78.5	91.2	—	90.2	96.7	—	86.1
DGMC	77.2	89.7	—	77.4	90.7	—	89.1	96.7	—	86.8
MultiKE	50.9	57.6	53.2	39.3	48.9	42.6	63.9	71.2	66.5	54.9
JarKA	70.6	87.7	76.6	64.6	85.5	70.8	70.4	88.8	76.8	76.8
SelfMEA	79.0	91.3	—	86.1	91.9	—	96.3	99.2	—	90.6
GEEA	76.1	94.6	82.7	75.5	95.3	82.7	77.6	96.2	84.4	85.0
UMAEA	80.0	96.2	86.0	80.1	96.7	86.2	81.8	97.3	87.7	88.0
MEAformer	77.0	96.1	84.1	76.4	95.9	83. 4	77.1	95.1	83.5	85.6
DESAlign	82.6	97.2	88.5	81.1	96.3	86.9	81.0	95.7	86.5	88.4
LLMEA	89.8	92.3	—	91.1	94.6	—	95.3	97.7	—	93.5
EAFF	85.7	97.0	95.8	87.7	97.7	95.7	95.3	98.7	96.5	94.5

EAFF相较于此前的研究,分别取得了0.2%~5.2%的提升。不可否认,在某些数据集的某些指标中,EAFF并没有达到最佳效果,但在大多数情况下,EAFF的各项指标都优于其他的基线模型的各项指标。EAFF的指标平均值(AVG)高于其他基线模型1.0%~30.1%。这证明了EAFF在实体对齐研究中的优良效果,也证明了图结构与元素信息进行融合对于实体对齐任务的促进作用。

4.3 总体结构消融实验

本实验旨在直观地展示图结构学习模块和元素信息学习模块对于实体对齐的影响。消融实验可以分为3组,Lab1消融掉图学习结构,Lab2消融掉元素信息学习结构,Lab3则作为原始对照组,不进行任何的结构消融。3组实验结果如表3所示。

表3 总体结构消融实验

语言名称	准确率					
	DBP_{ZH-EN}		DBP_{JA-EN}		DBP_{FR-EN}	
	Hits@1	Hits@10	Hits@1	EN	Hits@1	Hits@10
Lab1	83.1	95.4	82.5	93.7	95.3	98.7
Lab2	71.5	81.0	76.0	85.0	89.4	92.0
Lab3	85.7	97.0	87.7	93.7	95.3	98.7

上述3组实验,Lab1体现了元素信息对于实体对齐的促进效果,Lab2体现了图结构对于实体对齐的促进效果,Lab3体现了元素信息加图结构对于实体对齐的促进效果。对表3与图7进行分析,不难看出,在DBP15k的各项数据中,Lab3的数据普遍优于Lab1和Lab2,这证明,在EAFF中,图结构学习模块和元素信息学习模块对于实体对齐都起到了促进作用。

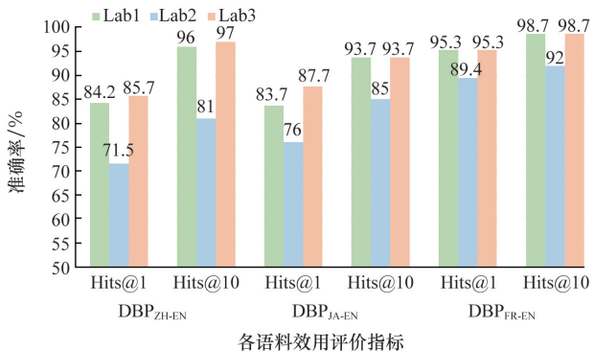

图7 消融实验结果

图8是对消融实验的3个实验组进行进一步的数据分析,可以发现,Lab1,即保留元素信息学习模

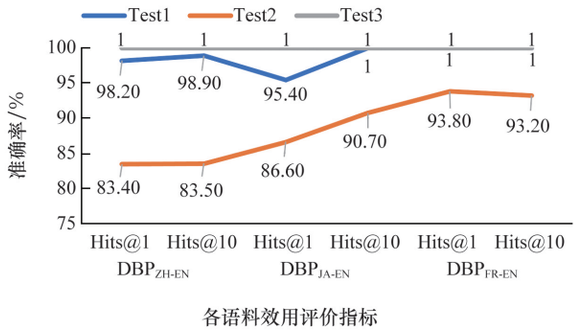

图8　消融实验效用对比

块的对照组的实体对齐准确率达到了原始模型的95.4%~100%;而Lab2,即保留图结构学习模块的对照组的实体对其准确率只有原始模型的83.4%~93.8%。通过对Lab1与Lab3、Lab2与Lab3的数据进行综合分析,可以得出结论:元素信息与结构特征对实体对齐都有促进作用;通过Lab1和Lab2对比,可以发现,图结构对于实体对齐的促进作用远逊于元素信息的促进作用。

值得注意的是,当元素信息结构将实体对齐达到一个相对较高的程度时,图结构对于元素信息结构的补充作用将大大降低。DBP$_{FR-EN}$的3组实验的结果证明了这一点:Lab1中只使用元素结构模型的Hits@1与Hits@10分别达到93.5%和98.7%时,Lab3中的原始模型的Hits@1与Hits@10也分别是93.5%和98.7%。由此可以确定,在DBP15k这类结构嵌入度较低的知识库中,当基于元素信息的对齐程度已经很高的前提下,图结构对于元素信息结构的补充作用将大大降低——或者说没起到任何的补充作用。

4.4　结构嵌入模块网络结构效用实验

本实验旨在证明结构嵌入模块网络结构的有效性。结构学习模块可以从图结构角度得到候选实体,本实验在DBP$_{ZH-EN}$上,测试不同的网络结构对于结构学习模块获取候选实体准确性的影响。Lab1为1层GAT的效果,Lab2为2层GAT的效果,Lab3为1层GCN的效果,Lab4为2层GCN的效果,Lab5为3层GCN的效果,Lab6为1层GAT加1层GCN的效果,Lab7为EAFF的原始网络结构,即1层GAT加2层GCN的效果。实验结果如表4所示。

表4　网络结构效用实验

评价指标	准确率/%						
	Lab1	Lab2	Lab3	Lab4	Lab5	Lab6	Lab7
Hits@1	47.76	46.26	54.69	56.73	55.47	69.01	71.46
Hits@10	69.80	69.19	68.44	69.31	68.76	78.94	80.97

通过表4与图9可以直观地证明,1层GAT加2层GCN的网络结构能相对最优地学习到知识图谱的图结构,并取得相对最好的实体对齐效果。通过对Lab2与Lab1和Lab5与Lab4这2组数据进行分析,可以发现网络层数并不是越深越好,这是因为网络的层数越深,其引入的噪声也就越大。此外,Lab6与Lab7的数据优于其他实验数据,这说明在知识图谱的结构分析中,GCN与GAT的结合效果优于单一网络结构的使用效果。这是因为GAT能够对有向图进行很好地学习,而GCN可以对完整的图结构进行很好地学习,二者的结合兼顾知识图谱中知识三元组的有向性与知识实体的稀疏性,因此,GCN与GAT结合使用能相对较好的学习知识图谱的图结构。

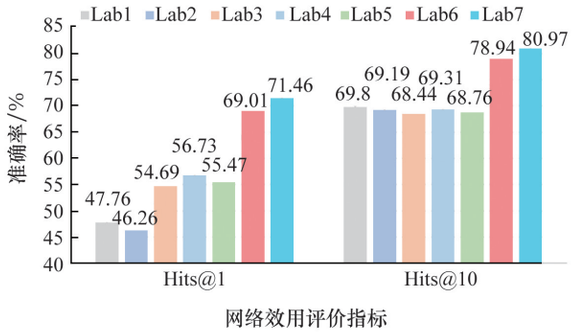

图9　网络结构效用对比

5　结论

设计了一种基于图结构与元素信息融合的实体对齐模型,该模型通过图结构学习模块和元素信息学习模块对知识图谱的图结构与元素信息进行学习,并利用基于排序思想的算法对学习到的特征进行融合。基线实验证明了本模型相较于此前研

究的提升,消融实验证明了本模型中的各个组件对实验效果的必要性。通过进一步的分析可以确定,未来实体对齐领域的研究将聚焦于图结构学习与元素信息学习的融合。

参考文献(References)

[1] Chen M H, Tian Y T, Yang M H, et al. Multilingual knowledge graph embeddings for cross-lingual knowledge alignment[EB/OL]. [2024-02-04]. http://arxiv.org/abs/1611.03954.

[2] Yang Y H, Huang C, Xia L H, et al. Knowledge graph self-supervised rationalization for recommendation[C]//Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. New York: ACM, 2023: 3046-3056.

[3] Liang K, Liu Y, Zhou S H, et al. Knowledge graph contrastive learning based on relation-symmetrical structure [J]. IEEE Transactions on Knowledge and Data Engineering, 2024, 36(1): 226-238.

[4] Fanourakis N, Efthymiou V, Kotzinos D, et al. Knowledge graph embedding methods for entity alignment: Experimental review[J]. Data Mining and Knowledge Discovery, 2023, 37(5): 2070-2137.

[5] Cao J H, Fang J Y, Meng Z Q, et al. Knowledge graph embedding: A survey from the perspective of representation spaces[J]. ACM Computing Surveys, 2024, 56(6): 1-42.

[6] Munne R F, Ichise R. Entity alignment via summary and attribute embeddings[J]. Logic Journal of the IGPL, 2023, 31(2): 314-324.

[7] Xu C H, Cheng J W, Zhang F. P-NAL: An effective and interpretable entity alignment method[EB/OL]. [2024-02-06]. http://arxiv.org/abs/2404.11968.

[8] Chen H J, Deng S M, Zhang W, et al. Neural symbolic reasoning with knowledge graphs: Knowledge extraction, relational reasoning, and inconsistency checking[J]. Fundamental Research, 2021, 1(5): 565-573.

[9] Jiang T T, Bu C Y, Zhu Y, et al. Combining embedding-based and symbol-based methods for entity alignment[J]. Pattern Recognition, 2022, 124: 108433.

[10] Liu X, Hong H Y, Wang X H, et al. SelfKG: Self-supervised entity alignment in knowledge graphs[C]//Proceedings of the ACM Web Conference 2022. New York: ACM, 2022: 860-870.

[11] Bordes A, Usunier N, Garcia-Duran A, et al. Translating embeddings for modeling multi-relational data[J]. Advances in neural information processing systems, 2013, 26: 1-9.

[12] Wang Z, Zhang J, Feng J, et al. Knowledge graph embedding by translating on hyperplanes[C]//Proceedings of the AAAI conference on artificial intelligence. California USA: Palo Alto, 2014: 1112-1119.

[13] Lin Y K, Liu Z Y, Sun M S, et al. Learning entity and relation embeddings for knowledge graph completion[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2015, 29(1): 2181-2187.

[14] Guo L B, Sun Z Q, Hu W. Learning to exploit long-term relational dependencies in knowledge graphs[EB/OL]. [2024-02-22]. http://arxiv.org/abs/1905.04914.

[15] Xiao H, Huang M L, Hao Y, et al. TransA: An adaptive approach for knowledge graph embedding[EB/OL]. [2024-02-22]. http://arxiv.org/abs/1509.05490.

[16] Shen L S, He R B, Huang S B. Entity alignment with adaptive margin learning knowledge graph embedding[J]. Data & Knowledge Engineering, 2022, 139: 101987.

[17] Li Y J, Tarlow D, Brockschmidt M, et al. Gated graph sequence neural networks[EB/OL]. [2024-02-22]. http://arxiv.org/abs/1511.05493.

[18] Mao X, Wang W, Xu H, et al. Relational reflection entity alignment[C]//Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2020: 1095-1104.

[19] Wu Y T, Liu X, Feng Y S, et al. Neighborhood matching network for entity alignment[EB/OL]. [2024-02-25]. http://arxiv.org/abs/2005.05607.

[20] Huang J, Wang J Q, Li Y H, et al. A survey of entity alignment of knowledge graph based on embedded representation[J]. Journal of Physics: Conference Series, 2022, 2171(1): 012050.

[21] Zeng W X, Zhao X, Tang J Y, et al. Collective entity alignment via adaptive features[C]//Proceedings of IEEE 36th International Conference on Data Engineering (ICDE). Piscataway, NJ: IEEE, 2020: 1870-1873.

[22] Yang H W, Zou Y Y, Shi P, et al. Aligning cross-lingual entities with multi-aspect information[EB/OL]. [2024-03-02]. http://arxiv.org/abs/1910.06575.

[23] Tang X, Zhang J, Chen B, et al. BERT-INT: A BERT-based interaction model for knowledge graph alignment [J]. interactions, 2020, 100: e1.

[24] Li Y J, Gu C J, Dullien T, et al. Graph matching net-

works for learning the similarity of graph structured objects[EB/OL]. [2024-03-15]. http://arxiv.org/abs/1904.12787

[25] Sun Z Q, Wang C M, Hu W, et al. Knowledge graph alignment network with gated multi-hop neighborhood aggregation[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(1): 222-229.

[26] Fey M, Lenssen J E, Morris C, et al. Deep graph matching consensus[EB/OL]. [2024-03-22]. http://arxiv.org/abs/2001.09621.

[27] Chen B, Zhang J, Tang X B, et al. JarKA: modeling attribute interactions for cross-lingual knowledge alignment [C]//Pacific-Asia Conference on Knowledge Discovery and Data Mining. Cham: Springer, 2020: 845-856.

[28] Liu B, Song R Y, Xiang Y J, et al. Self-supervised entity alignment based on multi-modal contrastive learning [J]. CAA Journal of Automatica Sinica, 2022, 9(11): 2031-2033.

[29] Guo L B, Chen Z, Chen J Y, et al. Revisit and outstrip entity alignment: A perspective of generative models[EB/OL]. [2024-04-22]. http://arxiv.org/abs/2305.14651.

[30] Chen Z, Guo L B, Fang Y, et al. Rethinking uncertainly missing and ambiguous visual modality in multi-modal entity alignment[C]//International Semantic Web Conference. Cham: Springer, 2023: 121-139.

[31] Chen Z, Chen J Y, Zhang W, et al. MEAformer: Multi-modal entity alignment transformer for meta modality hybrid[C]//Proceedings of the 31st ACM International Conference on Multimedia. New York: ACM, 2023: 3317-3327.

[32] Wang Y Y, Sun H F, Wang J B, et al. Towards semantic consistency: Dirichlet energy driven robust multi-modal entity alignment[EB/OL]. [2024-05-25]. http://arxiv.org/abs/2401.17859.

[33] Yang L Y, Chen H Y, Wang X, et al. Two heads are better than one: Integrating knowledge from knowledge graphs and large language models for entity alignment [EB/OL]. [2024-06-26]. http://arxiv.org/abs/2401.16960.

Entity alignment based on graph structure and element information fusion

MA Haoran, Wang Jinhua*

The 32nd Research Institute of China Electronics Technology Group Corporation, Shanghai 201808, China

Abstract Entity alignment, as an important research direction in knowledge graph research, aims to connect different entities pointing to the same real-world object in different knowledge graphs, and thus to achieve the expansion of knowledge graphs. At present, there are two mainstream research approaches in this field. One is to analyze the structural characteristics of knowledge graphs, and the other is to analyze the element information (such as entity name, relation name, attribute name) of knowledge graphs. In this article, a novel entity alignment model EAFF (Entity Alignment based on Feature Fusion) is proposed to analyze the features of knowledge graphs from the perspectives of graph structure and element information. First, a graph neural network-based entity alignment algorithm was designed to obtain aligned entity pairs based on graph structures. Then, an entity alignment algorithm based on element information was designed to obtain aligned entity pairs based on element information. Finally, using feature transformation and sorting algorithms, two sets of aligned entity pairs are sorted to obtain aligned entities in the knowledge graph. In the experiment, EAFF achieved relatively good results, surpassing current mainstream algorithms.

Keywords knowledge graph; entity alignment; graph neural network; feature fusion

（责任编辑　王微）