计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
2025,34(4):286−297 [doi: 10.15888/j.cnki.csa.009859] [CSTR: 32024.14.csa.009859]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

以表窥里: 聚焦表层信息的通用实体对齐方法①

郑百川, 陈　凯, 李升辉, 李冰倩, 张　宁

(华中科技大学 网络空间安全学院, 武汉 430062)
通信作者: 陈　凯, E-mail: kchen@hust.edu.cn

摘　要: 在知识图谱的整合过程中, 实体对齐 (EA) 任务至关重要. 最先进的研究引入了外部知识 (属性文本、时间戳、图像信息等) 以及多模态方法, 取得了较高的精度, 但这些方法往往对特定结构有较强的依赖性, 这限制了它们在不同结构知识图谱实体对齐任务中的适用性. 为了解决这一问题, 本文提出了一种通用的知识图谱实体对齐方法, 该方法利用知识图谱共有的实体、关系与图结构等信息工作, 上述部分在知识图谱中可被直接观察到, 因此统称为表层信息. 本文方法包含嵌入生成模块和对齐模块, 其中嵌入模块使用 Transformer 模型捕捉实体的固有语义及其邻居的贡献, 对齐模块则通过匹配算法实现高性能且稳定的对齐. 实验结果表明, 我们的方法在多个主流知识图谱间的对齐场景中实现了最先进的性能, 展现出稳定和可解释性强的特点. 我们的代码可在 https://github.com/zb1tree/TGEA 获取.

关键词: 知识图谱; 实体对齐; Transformer

引用格式: 郑百川,陈凯,李升辉,李冰倩,张宁.以表窥里: 聚焦表层信息的通用实体对齐方法.计算机系统应用,2025,34(4):286–297. http://www.c-s-a.org.cn/1003-3254/9859.html

Surface to Deeper: Universal Entity Alignment Approach Focusing on Surface Information

ZHENG Bai-Chuan, CHEN Kai, LI Sheng-Hui, LI Bing-Qian, ZHANG Ning

(School of Cyber Science and Engineering, Huazhong University of Science and Technology, Wuhan 430062, China)

Abstract: Entity alignment (EA) tasks are pivotal in the integration of knowledge graphs. The most advanced research has introduced external knowledge (attribute texts, timestamps, image information, etc.) and multimodal methods, achieving relatively high accuracy. However, these methods often have a strong dependence on specific structures, which limits their applicability in the entity alignment tasks of knowledge graphs with different structures. Therefore, this study proposes a universal knowledge graph alignment approach that utilizes the information of shared entity, relationship, and graph structure of knowledge graphs which are called surface information as they can be directly observed in knowledge graphs. An embedding generation module and an alignment module are included in the proposed method, and the former uses the Transformer model to capture the inherent semantics of entities and the contributions of their neighbors while the latter achieves high-performance and stable alignment through a matching algorithm. Experiment results show that the proposed method has achieved the best performance in the alignment scenarios among multiple mainstream knowledge graphs, demonstrating stability and strong interpretability. The code used in this study can be obtained at https://github.com/zb1tree/TGEA.

Key words: knowledge graph; entity alignment; Transformer

① 收稿时间: 2024-10-13; 修改时间: 2024-11-12; 采用时间: 2024-12-18; csa 在线出版时间: 2025-03-05
CNKI 网络首发时间: 2025-03-06

知识图谱的概念自 2012 年被谷歌提出以来, 始终受到广泛的关注. 随着人工智能技术的发展和应用, 知识图谱除了作为各类搜索引擎的数据库以外, 在智能检索、智能问答、知识推理与个性化推荐等领域都获得了广泛的应用.

随着知识图谱相关技术的发展, 很多机构都构建了自己的知识图谱, 由于数据来源, 构建目的和构建方法的不同, 这些知识图谱在结构上不可避免的是异质的, 并且包含的信息通常是互补的, 例如 DBpedia[1]、YAGO[2]、Freebase[3]. 知识图谱实体对齐旨在发现跨知识图谱的等效实体, 同时解决不同命名约定、不同语言和不同架构等挑战.

实体对齐相关研究致力于将跨知识图谱的实体映射到同一空间中, 通过距离判断实体间等效的可能性. 近年来, 新的工作集中在使用属性图等辅助信息协助发现等效实体, 以 MEAformer[4]为代表的多模态实体对齐 (MMEA) 方法使用包括文本描述与图像信息的多模态融合方法实现了至今为止最高的精准度, 但是这些方法严重依赖辅助信息, 同时忽略了辅助信息欠缺与知识图谱异构的现实应用场景.

在这项工作中, 我们提出了一种通用的知识图谱实体对齐方法, 可以在跨语言、跨架构的知识图谱上工作, 仅需要知识图谱共有的结构即可工作. 我们提出的方法包含嵌入生成模块和对齐模块, 嵌入模块使用 Transformer 捕捉实体的固有语义和来自邻居的贡献, 对齐模块使用匹配算法实现高性能且稳定的对齐. 综上所述, 本文的贡献如下.

• 总结了现有方法的局限性, 并提出了在实际应用场景中更加通用的实体对齐模型.

• 实现了一种基于 Transformer 和匹配算法的 EA 方法, 可以深入挖掘实体的表层与图结构包含的信息.

• 实验证明本文方法能够在多个主流知识图谱间的对齐场景实现 SOTA 性能, 结果稳定、可解释性强.

本文其余部分组织如下. 第 1 节介绍了相关工作的进展及其不足. 第 2 节给出了本文对 EA 问题的形式化定义. 第 3 节介绍了本文方法的实现细节. 第 4 节讨论实验结果.

1 相关工作

1.1 实体对齐

早期的实体对齐受到 Word2Vec[5]工作的启发, 假设实体嵌入之间可以利用关系嵌入进行运算. 在此思想指导下出现了实体对齐领域最具有代表性的模型之一 TransE[6], 该模型将跨知识图谱的实体映射到同一语义空间中, 通过非同源实体的距离判断其同质的可能性. 在此基础出现了一系列变体, 例如针对复杂关系处理的改进模型 TransH[7]、TransM[8]和 TransR[9], 挖掘实体间多跳关系的模型 PTransE[10], 使用极坐标的模型 RotatE[11]等.

神经网络模型一般以 trans 系列表示方法为基础, 进一步聚合图结构信息从而取得更好的效果. ConvE[12]通过将头实体和关系表示为二维向量, 利用多层图卷积获取交互信息, 结合尾实体定义评分函数以判断 (头, 关系, 尾) 三元组的真实性. 随着 GCN[13]研究的深入, 神经网络模型中出现了关系神经网络 (R-GCN)[14], 该模型在图卷积网络中通过特定关系转义来进行知识图谱嵌入. RDGCN[15]构建实体之间的关系图时引入对偶图的概念, 通过对偶图的限制增强对不同实体网络结构的辨别能力.

由于表层信息噪声大、异构性强, 在早期方法中, 表层信息一般使用 Word2Vec 处理, 生成启发式的初始嵌入, 其最终结果严重受限于初始化的精度. 然而, Transformer 的出现使得聚合异构信息生成统一实体表示成为可能. 基于语义的实体对齐方法[16]通过属性图蕴含的丰富文本信息生成实体嵌入, 多模态实体对齐[4]吸收包括图像信息的泛属性图取得了目前最高的对齐精度.

1.2 分配问题

分配问题又称指派问题, 是一种特殊的组合优化问题, 可以直观解释为: n 个工人需要完成 n 个任务, 每个任务只能分配给一个工人. 不同工人完成不同任务的成本不同, 需要得出一个分配方案使总成本最小. 形式上该问题可表示为求解式 (1):

$$\arg\min_{P\in P_N}\langle P,X\rangle_F \tag{1}$$

其中, P 是排列矩阵, X 是代价矩阵, $\langle\cdot\rangle_F$ 表示矩阵 Frobenius 内积. 解决分配问题最常用的方法是匈牙利算法, 该算法时间复杂度为$\mathrm{O}\left(n^3\right)$.

2 问题定义

我们将知识图谱定义为一个三元组的集合, 即 $\mathcal{G}=\{\mathcal{E},\mathcal{R},\mathcal{T}\}$, 其中$\mathcal{E}$, $\mathcal{R}$分别表示实体e (entity) 与关系

r (relation) 的集合. 实体是知识图谱中表示现实世界中具体对象或概念的节点, 它们可以是人、地点、组织、事件及概念等, 而知识图谱中的关系对应现实对象之间的关系, 在知识图谱中以实体间的边形式出现. $\mathcal{T} \subseteq \{\mathcal{E} \times \mathcal{R} \times \mathcal{E}\}$表示关系三元组的集合, 知识图谱中的每个三元组可被称为一条知识. 一般而言, 知识图谱中的关系存在指向性, 因此用e_h表示头实体, e_t表示尾实体, 一条知识的形式化表示为$t = \{e_h, r, e_t\}$. 给定两个知识图谱, $\mathcal{G}_1 = \{\mathcal{E}_1, \mathcal{R}_1, \mathcal{T}_1\}$ 和$\mathcal{G}_2 = \{\mathcal{E}_2, \mathcal{R}_2, \mathcal{T}_2\}$, 知识图谱实体对齐的目标是识别对应着相同现实世界对象的实体对$\left(e_1, e_1'\right) \subset KG_1 \times KG_2$, 其中$e_1 \in \mathcal{E}_1, e_1' \in \mathcal{E}_2$.

3 方法

3.1 挑战

知识图谱具有高度的模糊性和强烈的噪声, 而这必然对实体对齐造成影响. 图 1 是 DBP15K 数据集 ZH-EN 部分“李光耀”词条的局部, 我们将以此为例解释本文方法抗噪声的能力. 图 1 中红色为“李光耀”实体, 蓝色表示通过相同关系链接对应实体的尾实体, 黄色表示存在对应实体但是链接的关系不同的尾实体, 灰色表示无对应的尾实体.

1) 冗余: $\exists\left\langle e_{\{h_1\}}, r_1, e_{\{t_1\}}\right\rangle \cdots \left\langle e_{\{h_1\}}, r_m, e_{\{t_1\}}\right\rangle, \left\langle e_{\{h_1\}}, r_1, e_{\{t_n\}}\right\rangle \in \mathcal{G}_1$, 一个实体可能存在多个指向同一相邻实体的关系, 也存在对应同一关系的多个相邻实体, 在传统的嵌入生成模块中, 如果没有对关系的区分能力, 这种现象会导致混乱.

2) 混淆: $\exists\left\langle e_{\{h_1\}}, r_1, e_{\{t_1\}}\right\rangle, \left\langle e_{\{h_1\}}, r_2, e_{\{t_1\}}\right\rangle \in \mathcal{G}_1$, 其中$r_1 \approx r_2$, 存在意义相近的关系在本质相同的实体对之间被混淆使用, 如图 1 中的“successor”关系与“after”关系存在交叉, 因此在<李光耀, 吴作栋>实体对中被混用. 对于有区分关系能力的方法, 这种现象会导致有效信息缺失.

3) 时空重叠: $\exists\left\langle e_{\{h_1\}}, r_{t_1}, e_{\{t_1\}}\right\rangle, \left\langle e_{\{h_1\}}, r_{t_2}, e_{\{t_2\}}\right\rangle \in \mathcal{G}_1$, 其中, t_1和t_2表示不同的时间, $e_{\{t_1\}}$和$e_{\{t_2\}}$表示不同时间的尾实体. 知识图谱的基础结构中不包含时间戳, 因此会导致不同时间的知识堆叠在知识图谱中. 李光耀先后担任内阁资政和新加坡总理, 但是在中文数据库中未收录〈李光耀, 新加坡总理〉这一知识. 这种现象会导致混乱.

4) 空置: $\exists\left\langle e_{\{h_1\}}, r_1, e_{\{t_1\}}\right\rangle \in \mathcal{G}_1, \nexists\left\langle e_{\{h_1'\}}, r_1', e_{\{t_1'\}}\right\rangle \in \mathcal{G}_2$, 其中$e_{\{h_1'\}}, e_{\{t_1'\}}$对应于$e_{\{h_1\}}, e_{\{t_1\}}$, 知识图谱中实体的相邻实体并非一一对应, 因此必然会出现空置. 而传统方法不能判断邻居实体为空置的无效信息并将其丢弃, 大量的空置将会对结果造成影响.

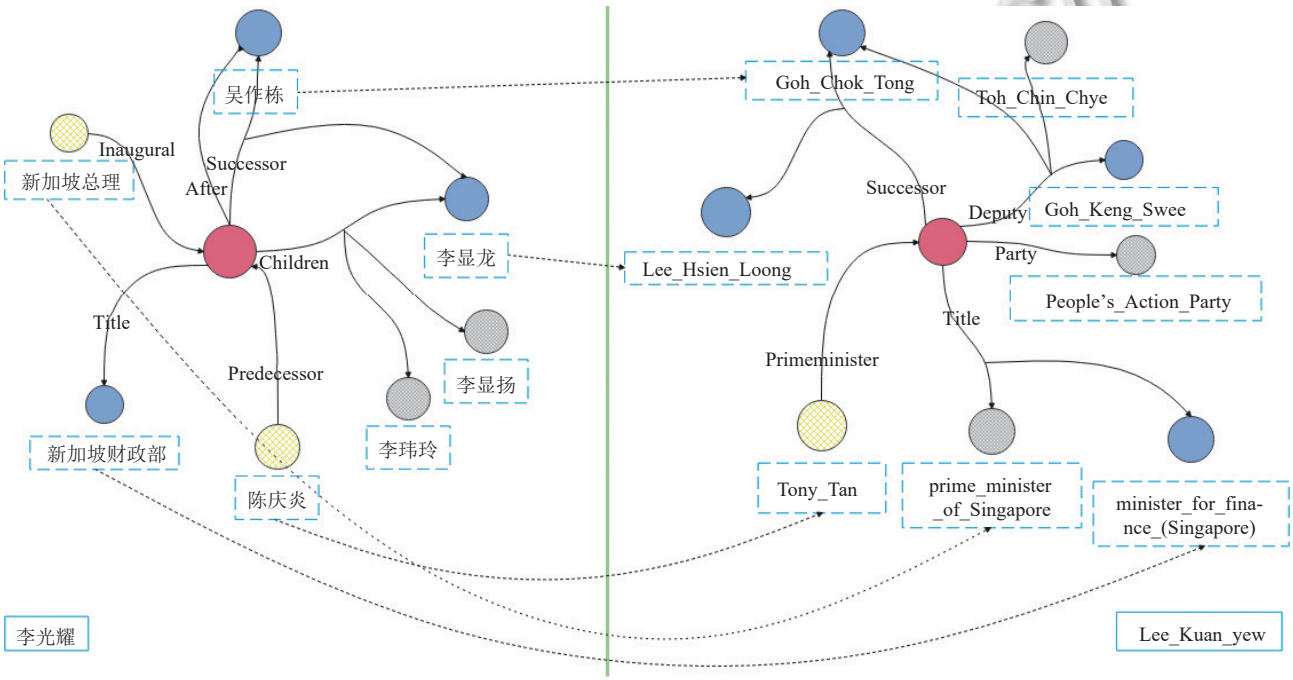

图 1 DBP15K ZH-EN 数据集“李光耀”词条的局部

3.2 设计

(1) 聚合相邻实体信息. 利用相邻实体信息的程度决定 EA 的效果上限, 基于以下观察到的现象, 我们设计了嵌入生成模块.

1) 反向关系. 定义关系r^-, 如果存在一种关系 r 使得(e_1,r,e_2)成立, 则(e_2,r^-,e_1)成立, 称r与r^-互为反向关系. 定义e_i的相邻实体集合: $\mathcal{N}_{e_i}=\{e_j|(e_i,r,e_j)\in\mathcal{G},r\in\mathcal{R}\}$. 如果利用反向关系对原本的知识图谱进行扩展, 则e_i的相邻实体定义扩展为: $\mathcal{N}_{e_i}=(e_j|(e_i,r,e_j)\in\mathcal{G}\vee(e_j,r,e_i)\in\mathcal{G},r\in\mathcal{R})$, 增加了实体可从知识图谱中聚合的信息量. 以图 1 中文知识图谱的“李光耀”词条为例, 该词条缺失了标志性的〈李光耀, *title*, 新加坡总理〉三元组, 这会在对齐时大大增加不确定性. 但是将反向关系纳入考虑, 我们观察到知识图谱中存在〈新加坡总理, *inaugural*, 李光耀〉三元组, 将其反向关系形式化表示为〈李光耀, *inaugural*$^-$, 新加坡总理〉, 这一三元组可以与英文知识图谱中的〈Lee_Kuan_yew, *title*, prime_minister_of_Singapore〉进行比较得出“李光耀”与“Lee_Kuan_yew”的相同本质. 如果正确处理反向关系, 可以增强对齐的稳定性.

2) 关系贡献. 我们注意到, 不同的关系链接到的邻居对实体对齐的贡献是有差异的. 关系可以大致分为两类:一般关系和特定关系. 一般关系往往链接到大量的实体, 代表了一类实体所共有的属性, 如人物实体的 *birthplace* 关系一般只能提供所在地信息, 链接到 9216 个实体, 而 *knownFor* 关系只链接到 638 个实体, 它可以提供职业、标志性成就等信息. 一般地, 我们认为一个关系链接的实体越少, 则其一般性往往越弱, 对特定实体对齐的贡献也就越强.

3) 近似语义. 知识图谱之间并不是严格同构的, 考虑到跨架构的知识图谱实体对齐, 将实体的邻居一一对应并不现实, 但是不对应的实体依然可能在语义上存在关联. 仍然以图 1 为例, 英文知识图谱中收录了 Lee_Kuan_yew 所在的党派是 People’s Action Party, 虽然在中文知识图谱中没有实体与之对应, 但它暗示了该实体对应现实中一名新加坡政治人物. 对于这类无法在图结构上直接对应, 但是在语义上对实体对齐有贡献的相邻实体, 我们希望在聚合相邻实体的过程中将其纳入考虑.

总的来说, 我们应当为不同的相邻实体赋予不同的重要性, 连接到相对特定关系的实体对实体对齐的贡献更大, 在嵌入生成时应当做出更多贡献. 相应的, 相对一般的关系连接到相邻实体应考虑其连接数量, 削弱它对嵌入的影响. 考虑到反向关系对实体对齐可能存在的影响, 在生成嵌入时应将反向关系与其他关系进行相同的处理. 由于近似语义的存在独立于结构上的相似性, 因此在聚合相邻实体信息时应当减少对图结构的依赖性. 为了充分地捕获语义, 我们选择基于 Transformer 设计嵌入生成模块, Transformer 已经被证明在捕获词之间的联系上拥有强大的效果. 为了量化不同相邻实体的贡献, 我们引入了注意力机制来学习不同相邻实体的贡献.

(2) 去除噪声. 在实体嵌入生成之后, 本质相同的实体从嵌入角度往往更接近. 但是鉴于知识图谱固有的模糊性, 即第 3.1 节初步的嵌入包含大量可能影响对齐准确性的噪声. 我们认为预对齐机制可以有效减少噪声, 提高最终对齐效果.

1) 知识差异. 我们将源知识图谱相对目标知识图谱的非同构部分称为噪声. 噪声有多种来源, 其中最主要的来源是不同知识图谱覆盖的现实世界知识存在差异. 例如图 1 中文知识图谱收录了李光耀的子女信息 (冗余), 但是忽略了他的职位信息 (空置), 并且两个知识图谱均将李光耀不同时间的信息一同收录 (时空重叠). 虽然在嵌入生成的过程中考虑语义信息可以一定程度上减少噪声, 但是不能完全避免信息差对嵌入产生的负面贡献.

2) 错误知识. 一个庞大的知识图谱往往包含了一定的错误知识. 如图 1 中文数据库包含了〈李光耀, *religious_reliefs*, 不可知论〉, 该词条并非现实世界的真实知识, 也没有被其他知识图谱收录, 因此会对实体对齐产生负面影响. 在知识图谱实体对齐的过程中, 知识的错误与空置邻居实体无法从本质上区分开, 我们必须考虑这种情况的存在并降低错误在实体嵌入中的负面贡献.

噪声在知识图谱中广泛存在, 且难以在数据层面识别去除噪声. 这类非真实三元组产生的效果接近于非目标对抗攻击产生的“负三元组”, 文献[17]指出负三元组对基于规则的实体对齐方法存在较大影响, 错误的嵌入可能在对齐的迭代过程中累计错误导致误差加大. 参考对抗攻击方法, 知识图谱中存在的错误等噪声

在大部分情况下可以视为微小的扰动. 在此基础上进行一次预对齐可以减少噪声在对齐过程中的干扰. 我们注意到实体对齐包含二部图与图节点一一对齐的特征, 与分配问题存在很大的相似性, 但是源知识图谱与目的知识图谱的异构性与分配问题存在冲突. 我们选择了改进的匈牙利算法, 这个算法不需要限制输入的两个图重构. 我们计算实体在嵌入空间中的欧氏距离, 实体间的欧氏距离越短则它们内蕴的现实世界对象相同的可能性越大.

(3) 嵌入对齐. 在经过预对齐后阶段后, 实体的嵌入产生了偏移. 直观地说, 部分指向相同现实对象的实体嵌入被对齐, 但是也存在部分实体被混淆. 在最终嵌入生成时, 需要将当前的实体嵌入与实体 relation 都纳入考虑, 区分被混淆实体.

我们选择了 GRU 捕获邻居信息并生成最终的实体嵌入, 若e_j是实体e_i的邻居, 则它对e_i的贡献受到所有的邻居影响. Bidirectional GRU (BiGRU)[18]可以同时处理正向和反向信息, 它能够提取到更多的特征信息, 这使得 BiGRU 在处理复杂任务时具有更强的特征提取能力. 在我们的问题中, BiGRU 模型可以捕获实体不同邻居的相关性, 根据上下文 (邻居) 为同一个邻居对不同实体的不同贡献进行建模.

3.3 系统结构

基于以上考虑, 我们提出了 Transformer-based generic entity alignment (TGEA). 如图 2 所示, 其中e与e'表示源图与目标图中对应现实对象相同的实体, 而$e-$表示与e本质不同的实体. TGEA 包括嵌入生成模块、预对齐模块和实体对齐模块. 我们的框架将为源图与目标图的实体生成最终的嵌入, 并作为对齐的基础. 在本节中, 我们将在第 3.3.1 节介绍实体的初始嵌入生成方法, 第 3.3.2 节介绍我们进行预对齐的方法, 在第 3.3.3 节中我们介绍聚合邻居信息并生成最终嵌入的方法.

3.3.1 嵌入生成模块

嵌入生成模块从知识图谱通用的表层信息中生成实体的初始嵌入, 在知识图谱模型中, 只有实体名称是通用的表层信息. 为了获取足够的信息量, 我们聚合邻居实体的名称作为生成初始嵌入的依据.

Transformer 对于实体名称的顺序并不敏感, 为规避知识图谱之间的异构性, 我们只需将实体名称按照相同规则进行排列形成一个整体, 由 Transformer 捕获其细粒度语义. 在我们的模型中, 我们将嵌入生成形式化为 Transformer 的代表性模型 BERT[19]的下游任务.

我们将实体 e 的邻居实体名称转换为序列, 该序列可以被视为一系列令牌送入 BERT 模型中. 不同关系在知识图谱中的出现频次有很大差异, 不妨将关系的出现频次称为普遍程度. 首先我们按照关系的普遍程度生成一个有顺序的关系集合$\hat{O}(\mathcal{R})=[r_1,r_2,\cdots,r_{|\mathcal{R}|}]$. 为了利用实体本身的名称, 我们将实体名视为一个关系, 相应的三元组为$(e,entity_name,e)$, 显然该关系是最为普遍的关系. $\mathcal{T}(e)$定义为全部以e为头实体的三元组的集合, 将其根据$\hat{O}(\mathcal{R})$的顺序组织为有序集合$\hat{\mathcal{T}}(e)$, 即$\hat{\mathcal{T}}(e)=\left[\left(e,r_1,e_{r_1}\right),\left(e,r_2,e_{r_2}\right),\cdots,\left(e,r_{|\mathcal{T}(e)|},e_{r_{|\mathcal{T}(e)|}}\right)\right]$. 最后我们提取$\hat{\mathcal{T}}(e)$中的尾实体名称形成实体$e$的有序邻居集合$\hat{\mathcal{E}}(e)=\left[e_1,e_2,\cdots,e_{|\hat{\mathcal{T}}(e)|}\right]$, 由该标记序列生成$e$的令牌序列$\mathcal{S}(e)$.

以图 1 所展示的英文局部为例, *title*、*successor*、*party*、 *primeminister* 和 *deputy* 对应的普遍程度分别为 11 592、6 436、2 083、1 713 和 940. 则该局部会以图 3 方式生成$\mathcal{S}$(Lee_Kuan_yew).

我们使用预训练的 BERT 模型和 MLP 层通过$\mathcal{S}(e)$生成实体e的初始嵌入. 首先在序列$\mathcal{S}(e)$的开头添加序列开端标记“[CLS]”, 得到符合 BERT 要求的输入序列$S'(e)$. 将该序列输入 BERT 得到实体的中间向量表示$C(e)$, 最后通过 MLP 获得初始向量表示e_{emb}. 形式上表示如下:

$$\mathcal{S}(e)=\text{“}e_{1,1}\cdots e_{1,m_1}\cdots e_{n,1}\cdots e_{n,m_n}\text{”} \tag{2}$$

$$\mathcal{S}'(e)=\text{“[CLS]”}\|\mathcal{S}(e) \tag{3}$$

$$C(e)=BERT(S'(e)) \tag{4}$$

$$H_s(e)=MLP(C(e)) \tag{5}$$

3.3.2 预对齐模块

预对齐模块基于嵌入生成模块产生的源图与目标图的实体嵌入进行一次最大似然的对齐. 此模块会寻找一个最佳的图级匹配, 从而使源图与目标图的对齐代价达到最小. 我们将寻找图级匹配形式化为分配问题, 为了简化表示, 我们假设$|\mathcal{E}_1|\leqslant|\mathcal{E}_2|$, 则任务可转化为寻找一个排列矩阵$P\in P_{|\mathcal{E}_1|}$, 使排列矩阵与代价矩阵的 Frobenius 内积最小.

$$\arg\min_{P\in p_{|\mathcal{E}_1|}} \langle P, X(\mathcal{E}_1,\mathcal{E}_2)\rangle \tag{6}$$

其中, $P_{|\mathcal{E}_1|}$表示源图的排列矩阵, $X(\mathcal{E}_1,\mathcal{E}_2)$表示源图与目标图的代价矩阵. 实体嵌入的对齐代价与基于嵌入的实体对齐方法一致, 使用欧氏距离计算:

$$X(i,j) = \|H_s(e_i) - H_s(e'_j)\|_2 \tag{7}$$

其中, $e_i \in \mathcal{E}_1, e'_j \in \mathcal{E}_2$.

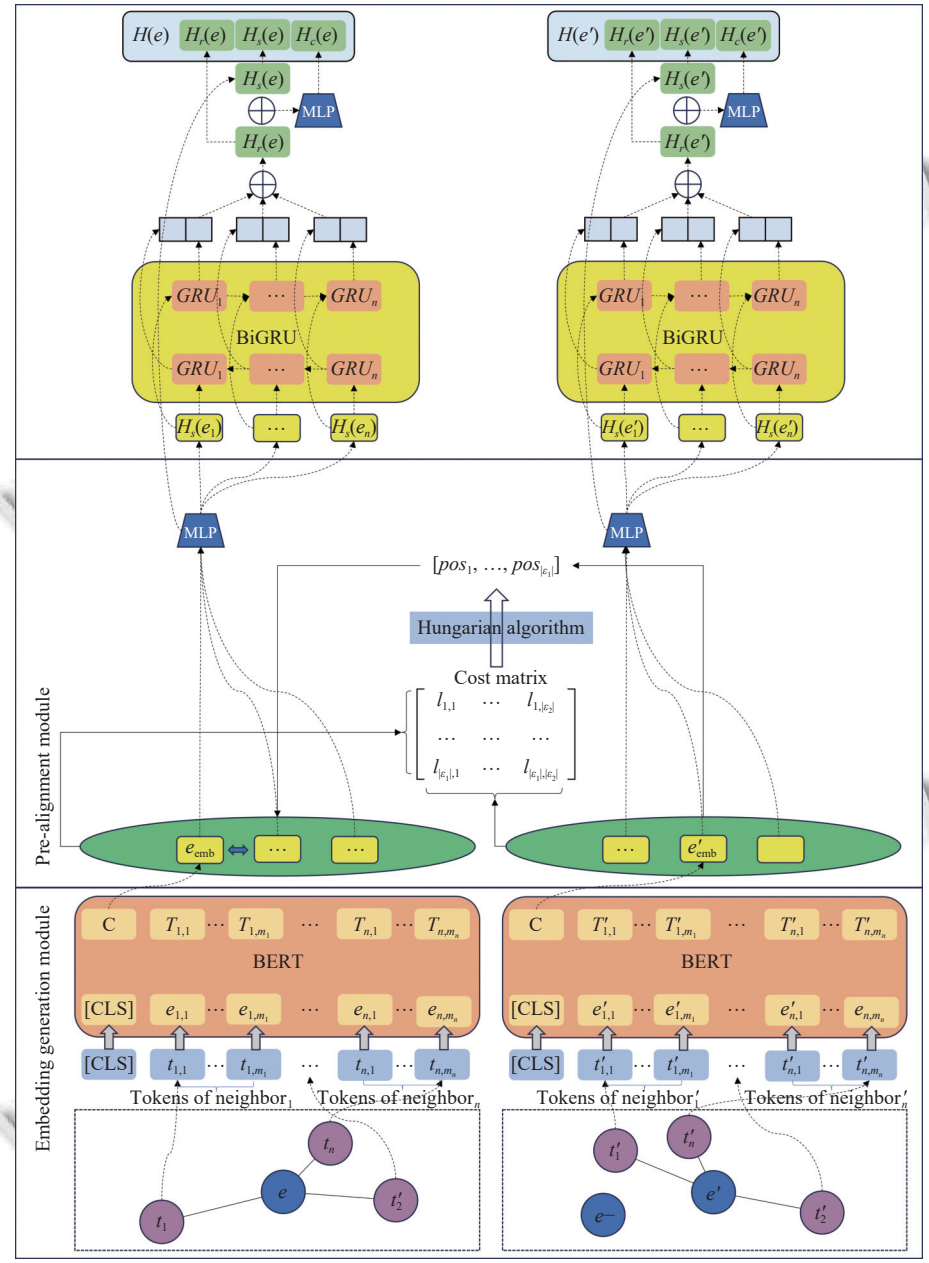

图 2 本文所提方法的框架

$e=\langle$Lee_Kuan_yew$\rangle$
$\hat{O}(e)=\langle entity\_name, title, successor, party, primeminister, deputy\rangle$
$\hat{\mathcal{T}}(\mathrm{e})=\langle$(Lee_Kuan_yew, $entity\_name$, Lee_Kuan_yew), $\cdots$, (Lee_Kuan_yew, $deputy$, Goh_Keng_Swee)$\rangle$
$\mathcal{S}(e)=$"Lee Kuan yew Prime Minister of Singapore $\cdots\cdots$"

图 3 令牌序列生成示例

算法 1 描述了我们进行预对齐的过程, 我们会依据最佳的图级匹配使源图与目标图的实体在嵌入空间中的投影保持一致. 首先根据源图与目标图的初始嵌入 计算对齐代价矩阵$cost_matrix$. 其中$cost_matrix(i,j)$表示源图实体e_i与目标图实体e'_j的对齐代价, 用欧氏距离计算:

$$cost\_matrix(i,j)=\|H_s(e_i)-H_s(e'_j)\|_2 \tag{8}$$

我们采用匈牙利算法得到满足式 (3) 的排列矩阵$marked_matrix$, 其每行及每列至多有 1 个被标记的元素, 若元素$marked_matrix(i,j)$被标记, 则表示e_i与e'_j是一对最大似然匹配.

算法 1. Hugary Pre—alignment algorithm

输入: $\mathcal{E}_1$: the set of embeddings of entities in $\mathcal{E}_1$; $\mathcal{E}_2$: the set of embeddings of entities in $\mathcal{E}_2$.

输出: E_1: the set of pre-aligned entities in $\mathcal{E}_1$; E_2: the set of pre-aligned entities in $\mathcal{E}_2$.

1. **for** $e_i \in \mathcal{E}_1$ **do**
2. **for** $e_j \in \mathcal{E}_2$ **do**
3. $cost\_matrix(ij)=\|e_i-e_j\|_2$
4. **end**
5. **end**
6. **step 1**:
7. **for** *row* ∈ *cost_matrix* **do**
8. find the smallest element and subtract it from every element in its *row*
9. **step 2**:
10. **for** *element* ∈ *cost_matrix* **do**
11. find a zero (*Z*) in the resulting matrix
12. **if** there is no starred zero in its row or column
13. star *Z*
14. **step 3**:
15. **for** *row* ∈ *cost_matrix* **do**
16. **if** the *row* contain a 0*
17. cover the *row*
18. **if** $|\mathcal{E}_1|$ rows are covered
19. **goto RESULT**
20. **step 4**:
21. find a noncovered zero and prime it
22. **if** no 0* in the row containing this primed zero
23. **goto step 5**
24. **else**
25. **for** exists uncovered zero **do**
26. cover this *row*
27. uncover the *row* containing the 0*
28. save the smallest uncovered value
29. **goto step 6**
30. **step 5**:
31. **for** a 0′ that has 0* inits column do
32. Let *Z*0 represent the uncovered 0′ found in **step 4**
33. Let *Z*1denote the 0* in the column of *Z*0
34. Let *Z*2 denote the 0′ in the row of *Z*1 (there will always be one)
35. unstar each 0*
36. star each 0′
37. erase all primes
38. uncover every line in the matrix
39. **goto step 3**
40. **step 6**:
41. add the value found in **step 4** to every element of each covered *row*
42. subtract it from every element of each uncovered column
43. **goto step 4**
44. **RESULT**:
45. $marked\_matrix(i,j)=\begin{cases}1, \text{if } cost\_matrix(i,j)=0^*\\ 0, \text{else}\end{cases}$
46. **for** $i\in|\mathcal{E}_1|$ **do**
47. append $e_j \in \mathcal{E}_2$ to E_1, where $marked\_matrix(i,j)$=1
48. $E_2=\mathcal{E}_2$

3.3.3 实体对齐模块

实体对齐模块为实体生成最终嵌入, 将源图与目标图实体映射到相同的向量空间, 并依据实体间的欧氏距离判断两个实体指向相同现实对象的可能性, 距离越短, 可能性越大. 下面我们将介绍生成最终嵌入和对齐的细节.

BiGRU 由两个方向的 GRU 网络组成, 可以同时处理正向和反向的信息. 其中每个 GRU 都包含一个重置门和一个更新门. 重置门控制忽略前一刻信息的程度, 更新门控制前一刻状态对当前状态的影响程度. 给定实体e, 记该实体的第t个邻居嵌入为x_t, 第t个隐藏单元的输出向量为h_t. 重置门r_t会丢弃对确定相关性不重要的邻居信息, 形式化表示为:

$$r_t=\sigma(W_r x_t+U_z h_{t-1}+b_r) \tag{9}$$

$$\tilde{h}_t=\tanh(W x_t)+U(r_t\odot h_{t-1}+b_h) \tag{10}$$

其中, W, U, b为可训练矩阵参数, σ表示 Sigmoid 函数, $\tilde{h}_t$是重置门产生的隐态. 更新门z_t会引入当前邻居嵌入的重要特征, 形式化表示为:

$$z_t=\sigma(W_z x_t+U_z h_{t-1}+b_z) \tag{11}$$

$$h_t=(1-z_t)\odot h_{t-1}+z_t\odot\tilde{h} \tag{12}$$

BiGRU 采用注意力机制识别邻居实体的重要性, 该部分以 BiGRU 的输出作为输入, 并输出一个权重向量, 权重向量中的每个元素表示对应邻居实体的贡献, 即注意系数. 给定实体的邻居实体嵌入, 注意力系数的

计算方法如下:

$$\alpha_t = \frac{\exp(w_t)}{\sum_{i=1}^{|\mathcal{N}(e)|}\exp(w_i)} \tag{13}$$

w_t表示邻居实体的贡献评分, 可以由下式计算:

$$w_t = h_t^{\mathrm{T}} \cdot \hat{h} \tag{14}$$

其中, $\hat{h}$由 BiGRU 的最后一个输出即$h_{|\mathcal{N}(e)|}$获得, 计算方法为:

$$\hat{h} = MLP(h_{\mathcal{N}(e)}) \tag{15}$$

我们的模型在聚合邻居实体信息后生成实体 e 的关系嵌入$h_r(e)$:

$$H_r(e) = \sum_{t=1}^{|\mathcal{N}_e|}\alpha_t \cdot h_t \tag{16}$$

我们额外计算了一个嵌入$H_c(e)$联合实体表层信息和邻居信息, 并将得到的 3 个嵌入联合作为实体的最终嵌入. $H_c(e)$的计算方法为:

$$H_c(e) = MLP(H_s(e)|H_r(e)) \tag{17}$$

基于表层信息, 关系信息以及表层信息与关系信息的间接关系的联合信息, 我们生成了最终的实体嵌入H_{ent}:

$$H_{\mathrm{ent}}(e) = H_r(e)|H_s(e)|H_c(e) \tag{18}$$

其中, |表示嵌入向量的连接. 我们将最终嵌入的欧氏距离视为实体的相似性, 并以此为依据为实体间对齐的可能性排序.

4 实验

我们进行了大量的实验, 并与最先进的方法进行比较. 在本节中, 我们将首先介绍我们的实验设置, 然后给出我们的实验结果和分析.

4.1 实验数据

为评估本文方法, 我们在两个广泛使用的数据集上进行了实验: DBP15K[20]和 SRPRS[21]. DBP15K 是最常用的跨语言 EA 数据集, 构建自 DBpedia. DBpedia 是世界上最大的多领域知识库之一, 在其庞大的知识内容和丰富的领域范围下, 也成为了谷歌, 雅虎等搜索引擎检索的支持. DBpedia 从维基百科的词条里抽取出结构化信息, 以加强维基百科的搜索能力, 并将其他知识库链接至维基百科. 这种结构化信息类似于一个开放的知识图谱, 可供互联网上的每个人使用. DBP15K 包含从 DBpedia 中提取的 3 个多语言数据集, 包括中文-英文 (ZH-EN)、日语-英文 (JA-EN) 和法语-英文 (FR-EN). 每个数据集包含 15 000 个跨语言链接 (对齐种子) 用于训练和测试. DBP15K 有两个版本, 完整版和精简版, 精简版是从完整版中提取得到的. 为了更方便地与已有研究进行比较, 我们选择精简版进行试验. SRPRS 也是一个被广泛使用的实体对齐基准数据集, 包含两个跨语言数据集 EN-DE 和 EN-FR, 以及两个跨架构数据集 DBP-WD (DW) 和 DBP-YG (DY). 其中跨语言数据集也是从 DBpedia 中提取, 跨架构数据集分别提取自 DBpedia、Wikipedia[22]和 YAGO. Wikipedia 的数据提取自 Wikidata, Wikidata 是一个免费、协作、多语言的数据库, 通过收集结构化数据以支持各种应用的使用. Wikidata 中的数据在发布之后, 允许在许多不同场景中复制、修改、分发, 甚至无需申请许可就用于商业目的. 其中的数据由维基数据编辑输入和维护, 他们决定内容创建和管理的规则. 并且数据是多语言的, Wikidata 鼓励使用任何语言进行编辑. YAGO 由德国马普研究所于 2007 年研制, 集成了维基百科、wordNet 和 GeoNames 这 3 个来源的数据, 是 IBM 沃森大脑的后端知识库之一. YAGO 利用规则对维基百科实体的 infobox 进行抽取, 通过实体类别推断构建“概念-实体”“实体-属性”间的关系. 表 1 展示了基准数据的参数.

表 1 基准数据

Datasets	Type	Language	Entities	Relations	Triples
DBP15K	ZH-EN	ZH	19 388	1 701	70 144
		EN	19 572	1 323	95 142
	JA-EN	JA	19 814	1 299	77 214
		EN	19 780	1 153	93 484
	FR-EN	FR	19 661	903	105 998
		EN	19 993	1 208	115 722
SRPRS	EN-FR	EN	15 000	221	36 508
		FR	15 000	177	33 532
	EN-DE	EN	15 000	222	38 363
		DE	15 000	120	37 377
	DBP-WD	DBP	15 000	253	38 421
		WD	15 000	144	40 159
	DBP-YG	DBP	15 000	223	33 748
		YG	15 000	30	36 569

4.2 评估指标

我们使用 *Hits*@1, *Hits*@10 以及 *MRR* (mean reciprocal ranking) 作为评价指标. *Hits*@*k* (*k*=1, 10) 准确率是用来计算预测结果中概率最大的前 *k* 个结果包含正确标签的占比, 形式化表示为:

$$Hits@k = \frac{\left|\{e_i|e_i \in \mathcal{E}_1 \wedge rank_i \leqslant k\}\right|}{|\mathcal{E}_1|} \tag{19}$$

其中, $rank_i$表示源实体e_i对应的目标实体在结果列表中的排位, 这个值越高越好. *MRR* 定义为结果列表中正确匹配的排位倒数的平均值, 形式化表示为式 (20), 这个值同样是越高越好.

$$MRR = \frac{1}{|\mathcal{E}_1|}\sum_{i=1}^{|\mathcal{E}_1|}\frac{1}{rank_i} \tag{20}$$

4.3 实验设置

在每个数据集中, 我们以 1:2:7 的比例划分验证集、训练集和测试集. 在实验中, 我们将 BERT 输入序列的最大长度固定为 128. 属性嵌入模块的批大小为 8, 关系嵌入模块的批大小为 256. 当验证集中的 *Hits*@1 连续 5 次不增加时, 训练过程终止.

4.4 比较方法

我们对截至 2022 年以 DBP15k 和 SRPRS 为基准的方法进行了总结, 并基于此评估我们的方法. 为了增加挑战性, 我们的对比对象中包含了多个多模态实体对齐 (MMKG) 方法, 包括 SDEA[16]、BERT-INT[23]、EVA[17]、MSNEA[24]和 MEAformer[4], 旨在评估我们的方法在保证通用性的前提下是否可以接近特定领域方法的准确度, 为公平起见, MMEA 方法的结果不参与结果比较. 这种方法同样使用 Transformer 生成原始嵌入, 并且利用了两个主流基准配套的属性图, 在实体对齐领域取得了显著优于传统方法的效果.

4.5 实验结果

本节给出了我们的实验结果. 在表 2 中我们给出了在 DBP15K 数据集上的实验结果并与其他 EA 方法进行了比较, 并且提供了一种典型的 MMEA 方法的实验结果作为对比. 在表 3 中我们给出了在更有挑战性的 SRPRS 数据集上的实验结果, 同样给出了与其他 EA 方法及 MMEA 方法的对比.

表 2 在 DBP15K 上的实验结果

类别	方法	ZH-EN			JA-EN			FR-EN		
		Hits@1	*Hits*@10	*MRR*	*Hits*@1	*Hits*@10	*MRR*	*Hits*@1	*Hits*@10	*MRR*
Trans series	MTransE	20.9	51.2	0.31	25	57.2	0.36	24.7	57.7	0.36
	JAPE-Stru	37.2	68.9	0.48	32.9	63.8	0.43	29.3	61.7	0.4
	JAPE	41.4	74.1	0.53	36.5	69.5	0.48	31.8	66.8	0.44
	NANE	38.5	63.5	0.47	35.3	61.3	0.44	30.8	59.6	0.4
	BootEA	61.4	84.1	0.69	57.3	82.9	0.66	58.5	84.5	0.68
	TransEdge	75.3	92.4	0.81	74.6	92.4	0.81	77	94.2	0.83
Long-term dependency	IPTransE	33.2	64.5	0.43	29	59.5	0.39	24.5	56.8	0.35
	RSN4EA	58	81.1	0.66	57.4	79.9	0.65	61.2	84.1	0.69
GCN based	GCN	39.8	72	0.51	40	72.9	0.51	38.9	74.9	0.51
	GCN-Align	43.4	76.2	0.55	42.7	76.2	0.54	41.1	77.2	0.53
	MuGNN	47	83.5	0.59	48.3	85.6	0.61	49.1	86.7	0.62
	KECG	47.7	83.6	0.6	49.2	84.4	0.61	48.5	84.9	0.61
	HMAN	56.1	85.9	0.67	55.7	86	0.67	55	87.6	0.66
	RDGCN	69.7	84.2	0.75	76.3	89.7	0.81	87.3	95	0.9
	HGCN	70.8	84	0.76	75.8	88.9	0.81	88.8	95.9	0.91
Literal	CEA	71.9	85.4	0.77	78.5	90.5	0.83	92.8	98.1	0.95
	TGEA (Ours)	88.9	95.9	0.91	83.3	92.9	0.86	98.3	99.7	0.99
	w/o pa.	83.3	94.4	0.87	80.4	92.2	84.6	96.2	99.2	0.97
MMEA	BERT-INT	81.4	83.7	0.82	80.6	83.5	0.82	98.7	99.2	0.99
	MSNEA	85.8	93.5	0.89	92.1	97.3	0.93	95.3	99	0.97
	EVA	88.3	96.7	0.91	93	98.5	0.97	96.8	99.5	0.97
	SDEA	87	96.6	0.91	84.8	95.2	0.89	96.9	99.5	0.98
	MEAformer	94.9	99.3	0.96	97.8	99.9	0.98	99.1	100	1

(1) 基线. 我们将 16 个基线按照技术路线分为 4 组, 第 1 组是 Trans 系列, Trans 系列通常基于翻译模型将实体名称映射到同一个嵌入空间, 并且基于嵌入进行实体对齐. 这类方法将实体与关系的嵌入解释为可进行几何计算的矢量, 如 TransE 方法认为三元组对应(h,r,t)在嵌入空间的中的关系为$\vec{h}+\vec{r}=\vec{t}$. 其中$\vec{h}$、$\vec{r}$、$\vec{t}$为头实体、关系、尾实体对应的嵌入向量. MTransE[25]、JAPE[20]与 NAEA[26]均为 TransE 方法的变体, 但是研究[15]指出这类方法对复杂关系 (如一对多、循环等) 难以处理, 因此对齐精度较低. BootEA[27]采用了 Boot-

strap 策略获取更优的嵌入, TransEdge[28]方法采用半监督方法优化了对复杂关系的处理, 取得了相对较好的结果.

第 2 组是基于长期依赖的方法, 这类方法在 Trans 系列的基础上通过路径预测实体间的对应关系, 但是这类方法并不能解决本文提到的各类挑战, 无法达到较为理想的结果.

第 3 组是基于图神经网络的方法, 与 Trans 系列类似, 同样把实体映射到嵌入空间, 但是并不对实体与关系进行计算而实聚合多种信息寻找准确的嵌入. GCN[29]与 GCN-Align[29]均只考虑了实体间的连接而不考虑连接对应的关系类型, 因此效果较差. MuGCN[30]、KECG[31]和 HMAN[32]都尝试捕获异构图中的实体与关系信息, 取得了较好的性能. RDGCN 构建了对偶图将实体与关系分别视为点与边, 并且进行交叉迭代从而充分利用图信息. HMAN 通过在不同层次上进行图卷积操作, 逐步聚合和更新节点的表示, 从而学习到更丰富的实体特征.

第 4 组利用了实体的表层特征, CEA[33]使用多种技术从图结构与语义中获取信息. 我们引入的 MMEA 方法同样是利用了语义信息, 但是除了通用的表层信息外, 这两个方法还是用了属性图提供的额外语义强化对齐效果.

(2) 结果分析. 表 2 显示我们的方法在 DBP15K 中的所有指标都优于传统方法, 在 *Hits*@1 指标上达到了 5%–17% 不等的提升, 与 MMEA 方法相当.

表 3 在 SRPRS 上的实验结果

类别	方法	EN-FR			EN-DE			D-W			D-Y		
		Hits@1	*Hits*@10	*MRR*	*Hits*@1	*Hits*@10	*MRR*	*Hits*@1	*Hits*@10	*MRR*	*Hits*@1	*Hits*@10	*MRR*
Trans series	MTransE	21.3	44.7	0.29	10.7	24.8	0.16	18.8	38.2	0.26	19.6	40.1	0.27
	JAPE-Stru	24.1	53.3	0.34	30.2	57.8	0.30	21.0	48.5	0.30	21.5	51.6	0.32
	JAPE	24.1	54.4	0.34	26.8	54.7	0.31	21.2	50.2	0.31	19.3	50.0	0.30
	NANE	17.7	41.6	0.26	30.7	53.5	0.26	18.2	42.9	0.26	19.5	45.1	0.28
	BootEA	36.5	64.9	0.46	50.3	73.2	0.48	38.4	66.7	0.48	38.1	65.1	0.47
	TransEdge	40.0	67.5	0.49	55.6	75.3	0.63	46.1	73.8	0.56	44.3	69.9	0.53
Long_term dependency	IPTransE	12.4	30.1	0.18	13.5	31.6	0.20	10.1	26.2	0.16	10.3	26.0	0.16
	RSN4EA	35.0	63.6	0.44	48.4	72.9	0.57	39.1	66.3	0.48	39.3	66.5	0.49
GCN based	GCN	24.3	52.2	0.34	38.5	60.0	0.46	29.1	55.6	0.38	31.9	58.6	0.41
	GCN-Align	29.6	59.2	0.40	42.8	66.2	0.51	32.7	61.1	0.42	34.7	64.0	0.45
	MuGNN	13.1	34.2	0.20	24.5	43.1	0.31	15.1	36.6	0.22	17.5	38.0	0.24
	KECG	29.8	61.6	0.40	44.4	70.7	0.54	32.3	64.6	0.43	35.0	65.1	0.45
	HMAN	40.0	70.5	0.50	52.8	77.8	0.62	43.3	74.4	0.54	46.1	76.5	0.56
	RDGCN	67.2	76.7	0.71	77.9	88.6	0.82	97.4	99.4	0.98	99.0	99.7	0.99
	HGCN	67.0	77.0	0.71	76.3	86.3	0.80	98.9	99.9	0.99	99.1	99.7	1.00
Literal	CEA	93.3	97.4	0.95	94.5	98.0	0.96	99.9	1.0	1.00	99.9	1.0	1.00
	TGEA (Ours)	97.8	98.7	0.98	97.1	98.3	0.98	82.8	87.8	0.85	99.9	1.0	1.00
	w/o pa.	93.8	97.0	0.95	96.4	98.6	0.97	63.2	72.7	0.66	99.9	1.0	1.00
MMEA	BERT-INT	97.1	97.5	0.97	98.6	98.8	0.99	99.6	99.7	1.00	100.0	100.0	1.00
	MSNEA	87.0	94.5	0.90	89.6	96.9	0.92	94.2	98.6	0.96	97.1	99.8	0.98
	EVA	93.7	99.1	0.96	95.6	99.3	0.97	97.9	99.8	0.99	99.5	99.9	1.00
	SDEA	96.6	98.6	0.97	96.8	98.9	0.98	98.0	99.6	0.99	99.9	1.0	1.00
	MEAformer	96.2	99.8	0.98	97.3	99.8	0.98	99.1	100	1.00	99.6	100	1.00

表 3 显示本文方法在 4 个数据集中的 3 个上效果领先于传统方法. 值得注意的是, SRPRS 数据集相比 DBP15K 更加稀疏, 因此 Trans 系列方法、基于长效依赖的方法与基于 GCN 的方法均产生了严重的下滑, 意味着这些方法对图结构信息有着较强的依赖. 与此相反, 基于语义的方法效果显著提升, 这体现了基于表层信息方法在处理信息不足时的稳定性, 同时也暗示这些方法在稠密的知识图谱下更容易受到噪声干扰.

(3) 消融实验. 在表 2 和表 3 中, 我们用 w/o pa. 展示了消融实验结果. 相比完整版本, w/o pa. 版本消除了预对齐模块, 结果表明在没有预对齐模块的情形下, 本文方法依然在大部分数据集中领先于基线. 在加入预对齐模块后, 我们的结果几乎都获得了进一步提升.

5 结语

本文中深入探讨了知识图谱实体对齐的挑战, 并

提出了一种创新的通用方法——Transformer-based generic entity alignment (TGEA). 该方法通过利用知识图谱共有的基础结构, 不依赖于特定的属性信息, 有效地解决了跨语言和跨架构的知识图谱对齐问题. 本文方法包含嵌入生成模块、预对齐模块和实体对齐模块, 其中嵌入模块利用 Transformer 模型捕捉实体的固有语义及其邻居的贡献, 预对齐模块通过改进的匈牙利算法减少噪声干扰, 而实体对齐模块则通过 BiGRU 和注意力机制生成最终的实体嵌入.

通过在主流知识图谱间的对齐场景中的实验, 我们证明了 TGEA 方法在多个评估指标上均实现了先进性能, 展现了其在不同数据集上的稳定性和可解释性. 此外, 消融实验进一步验证了预对齐模块在提升对齐效果中的重要作用.

总的来说, TGEA 方法为知识图谱的实体对齐任务提供了一种新的视角和解决方案, 其通用性和高效性使其在实际应用中具有广泛的价值. 我们期望这一研究能够推动知识图谱融合技术的发展, 并为未来的研究者提供新的思路和工具.

参考文献

1 Lehmann J, Isele R, Jakob M, *et al*. DBpedia—A large-scale, multilingual knowledge base extracted from Wikipedia. Semantic Web, 2015, 6(2): 167–195. [doi: 10.3233/SW-140134]

2 Rebele T, Suchanek F, Hoffart J, *et al*. YAGO: A multilingual knowledge base from Wikipedia, WordNet, and Geonames. Proceedings of the 15th International Semantic Web Conference. Kobe: Springer, 2016. 177–185.

3 Bollacker K, Evans C, Paritosh P, *et al*. Freebase: A collaboratively created graph database for structuring human knowledge. Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data. Vancouver: ACM, 2008. 1247–1250.

4 Chen Z, Chen JY, Zhang W, *et al*. MEAformer: Multi-modal entity alignment Transformer for meta modality hybrid. Proceedings of the 31st ACM International Conference on Multimedia. Ottawa: ACM, 2023. 3317–3327.

5 Mikolov T, Chen K, Corrado G, *et al*. Efficient estimation of word representations in vector space. arXiv:1301.3781, 2013.

6 Bordes A, Usunier N, Garcia-Durán A, *et al*. Translating embeddings for modeling multi-relational data. Proceedings of the 27th International Conference on Neural Information Processing Systems. Lake Tahoe: Curran Associates Inc., 2013. 2787–2795.

7 Wang Z, Zhang JW, Feng JL, *et al*. Knowledge graph embedding by translating on hyperplanes. Proceedings of the 28th AAAI Conference on Artificial Intelligence. Québec City: AAAI, 2014. 1112–1119.

8 Fan M, Zhou Q, Chang E, *et al*. Transition-based knowledge graph embedding with relational mapping properties. Proceedings of the 28th Pacific Asia Conference on Language, Information and Computation. Waseda University, 2014. 328–337.

9 Moon C, Jones P, Samatova NF. Learning entity type embeddings for knowledge graph completion. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. Singapore: ACM, 2017. 2215–2218.

10 Lin YK, Liu ZY, Luan HB, *et al*. Modeling relation paths for representation learning of knowledge bases. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. Lisbon: ACL, 2015. 705–714.

11 Sun ZQ, Deng ZH, Nie JY, *et al*. Rotate: Knowledge graph embedding by relational rotation in complex space. Proceedings of the 7th International Conference on Learning Representations. New Orleans: OpenReview.net, 2019.

12 Dettmers T, Minervini P, Stenetorp P, *et al*. Convolutional 2D knowledge graph embeddings. Proceedings of the 32nd AAAI Conference on Artificial Intelligence. New Orleans: AAAI, 2018. 1811–1818.

13 Liu X, Xia GQ, Lei FY, *et al*. Higher-order graph convolutional networks with multi-scale neighborhood pooling for semi-supervised node classification. IEEE Access, 2021, 9: 31268–31275. [doi: 10.1109/ACCESS.2021.3060173]

14 Schlichtkrull M, Kipf TN, Bloem P, *et al*. Modeling relational data with graph convolutional networks. Proceedings of the 15th International Conference. Heraklion: Springer, 2018. 593–607.

15 Wu YT, Liu X, Feng YS, *et al*. Relation-aware entity alignment for heterogeneous knowledge graphs. arXiv:1908.08210, 2019.

16 Zhong ZY, Zhang MH, Fan J, *et al*. Semantics driven embedding learning for effective entity alignment. Proceedings of the 38th IEEE International Conference on Data Engineering. Kuala Lumpur: IEEE, 2022. 2127–2140.

17 Liu FY, Chen MH, Roth D, *et al*. Visual pivoting for (unsupervised) entity alignment. Proceedings of the 35th

AAAI Conference on Artificial Intelligence. AAAI, 2021. 4257–4266.

18 Cho K, Van Merriënboer B, Gulcehre C, *et al*. Learning phrase representations using RNN encoder-decoder for statistical machine translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. Doha: ACL, 2014. 1724–1734.

19 Devlin J, Chang MW, Lee K, *et al*. BERT: Pre-training of deep bidirectional Transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics. Minneapolis: ACL, 2019. 4171–4186.

20 Sun ZQ, Hu W, Li CK. Cross-lingual entity alignment via joint attribute-preserving embedding. Proceedings of the 16th International Semantic Web Conference. Vienna: Springer, 2017. 628–644.

21 Guo LB, Sun ZQ, Hu W. Learning to exploit long-term relational dependencies in knowledge graphs. Proceedings of the 36th International Conference on Machine Learning. Long Beach: PMLR, 2019. 2505–2514.

22 Völkel M, Krötzsch M, Vrandecic D, *et al*. Semantic Wikipedia. Proceedings of the 15th International Conference on World Wide Web. Edinburgh: ACM, 2006. 585–594.

23 Tang XB, Zhang J, Chen B, *et al*. BERT-INT: A BERT-based interaction model for knowledge graph alignment. Proceedings of the 29th International Joint Conference on Artificial Intelligence. Yokohama: ijcai.org, 2020. 3174–3180.

24 Chen LY, Li Z, Xu T, *et al*. Multi-modal siamese network for entity alignment. Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. Washington: ACM, 2022. 118–126.

25 Chen MH, Tian YT, Yang MH, *et al*. Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. Proceedings of the 26th International Joint Conference on Artificial Intelligence. Melbourne: AAAI, 2017. 1511–1517.

26 Zhu QN, Zhou XF, Wu J, *et al*. Neighborhood-aware attentional representation for multilingual knowledge graphs. Proceedings of the 28th International Joint Conference on Artificial Intelligence. Macao: ijcai.org, 2019. 1943–1949.

27 Sun ZQ, Hu W, Zhang QH, *et al*. Bootstrapping entity alignment with knowledge graph embedding. Proceedings of the 27th International Joint Conference on Artificial Intelligence. Stockholm: ijcai.org, 2018. 4396–4402.

28 Sun ZQ, Huang JC, Hu W, *et al*. Transedge: Translating relation-contextualized embeddings for knowledge graphs. Proceedings of the 18th International Semantic Web Conference. Auckland: Springer, 2019. 612–629.

29 Xu K, Wang LW, Yu M, *et al*. Cross-lingual knowledge graph alignment via graph matching neural network. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence: ACL, 2019. 3156–3161.

30 Cao YX, Liu ZY, Li CJ, *et al*. Multi-channel graph neural network for entity alignment. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence: ACL, 2019. 1452–1461.

31 Li CJ, Cao YX, Hou L, *et al*. Semi-supervised entity alignment via joint knowledge embedding model and cross-graph model. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong: ACL, 2019. 2723–2732.

32 Yang HW, Zou YY, Shi P, *et al*. Aligning cross-lingual entities with multi-aspect information. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong: ACL, 2019. 4431–4441.

33 Wolf T, Debut L, Sanh V, *et al*. Transformers: State-of-the-art natural language processing. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. ACL, 2020. 38–45.

(校对责编: 王欣欣)