# InterTris:三元交互的领域知识图谱表示学习

张 祎 孟小峰
(中国人民大学信息学院 北京 100872)

**摘 要** 在新事物不断涌现,且事物之间联系不断丰富的时代背景下,作为一项新生技术,知识图谱旨在对现实世界中概念或实体及其之间的联系进行建模.由于直接来自于现实世界,知识图谱中的实体和关系往往以符号化形式表示.要实现进一步的价值挖掘,进行知识图谱计算,就需要将符号化表示转换为数值形式.知识图谱表示学习技术应运而生.目前,知识图谱表示学习已得到很大发展.依据应用领域不同,可以将知识图谱划分为通用领域和特定领域两种.已有表示学习模型多面向通用领域构建,且在通用领域的样本数据上进行验证.如果将这些模型运用到特定领域,就会面临新的数据分布挑战.为解决特定领域的知识图谱表示学习问题,本文以栖息地知识图谱和用户消费行为知识图谱为例进行了数据特征分析,发现特定领域知识图谱的数据特征不仅与通用领域不同,且不同领域之间的分布也各有特点.所以,我们从比数据分布更抽象的角度,即基于知识图谱构建语义联系的本质特征,以三元组为建模粒度,对头实体、关系和尾实体之间的交互作用进行了充分拟合,提出 InterTris 模型.同时,基于家谱领域的公共知识图谱 Kinship、微生物领域的酶知识图谱样本 ES、微生物领域的栖息地知识图谱样本 LiveIn 和电子商务领域的用户消费行为知识图谱样本 UserAct 共计四个数据集,以部分较优的转换模型和组合模型为基线,通过链接预测和三元组分类两组实验,本文发现 InterTris 在四个数据集上都取得了整体最优的效果,充分证明了在三元组粒度进行交互建模的必要性和合理性.

**关键词** 知识图谱;表示学习;特定领域;三元交互;链接预测;三元组分类
**中图法分类号** TP18 **DOI 号** 10.11897/SP.J.1016.2021.01535

# InterTris: Specific Domain Knowledge Graph Representation Learning by Interaction Among Triple Elements

ZHANG Yi MENG Xiao-Feng
(*School of Information, Renmin University of China, Beijing* 100872)

**Abstract** In the Big Data Era, new things are constantly arising and the connections among things are also constantly enriched, plenty of new technologies were born. As one of them, the emerging knowledge graph aims to describe entities or concepts in the real world and the connections among them. Different from traditional ways, it is a new organization, management and application way for large-scale data. So far, knowledge graph has played an important role in knowledge question answering, recommendation system, machine translation and so forth. To achieve further value mining from knowledge graph via computing, it is necessary to make full use of data mining technologies like machine learning, whose input are mainly numerals. However, based on the real world, elements in knowledge graph are all represented in symbolized form. So, the original symbolized representation needed to be converted into numerical one. This is the reason why knowledge graph representation learning was born. Up to now, after development of nearly 10 years, knowledge graph representation learning has made great progress, including

收稿日期:2019-09-05;在线发布日期:2020-02-10.本课题得到国家自然科学基金(91846204,61941121)资助.**张 祎**,硕士研究生,主要研究方向为知识图谱构建和知识图谱表示学习.E-mail:yizhang1208@ruc.edu.cn.**孟小峰**(通信作者),教授,博士生导师,中国计算机学会(CCF)会员,主要研究领域为大数据融合(知识图谱构建)、大数据实时分析、大数据隐私管理以及社会计算等交叉性研究.E-mail:xfmeng@ruc.edu.cn.

translation models, composition models and neural network models. According to the application domain, knowledge graph can be divided into open and specific domain. Freebase, YAGO, WikiData, DBpedia and Nell are all open domain ones. As for the specific domain, they focus on scientific research, e-commerce and so on. Based on open domain knowledge graph, the existing representation learning models are constructed and verified. On the one hand, semantics in specific domains are more concentrated. It has different data distribution from the open domain, especially in given application. On the other hand, the existing models are constructed based on particular data features. So they rely on the given data features a lot, preventing further applications. Therefore, if applied in specific domains, these models will be challenged by new data distributions. To deal with the knowledge graph representation learning problem in specific domains, taking the habitat knowledge graph and the customers behaviors knowledge graph as examples, we analyzed their features and found that there exist differences not only between open and specific domain but also among various specific domains. Therefore, from the perspective of more abstract than data distribution, based on the semantic connection construction essence of knowledge graph, this paper took a triplet as the granularity. At the same time, to make sure the features completion, considering the fact that any element among head entity, relation and tail entity is affected by the other two, this paper put forward InterTris by modeling the interaction among head entity, relation and tail entity. After that, taking some better translation and composition models as baselines, based on the public knowledge graph Kinship in genealogy, the enzyme knowledge graph sample ES in microbiology, the habitat knowledge graph sample LiveIn in microbiology, and the consumer behaviors knowledge graph sample UserAct in e-commerce, this paper carried out two experimental tasks, i.e. link prediction and triplets classification. Although not constructed for given data features, these experiments showed that InterTris has the best overall effect, proving the necessity and rationality of the triplet granularity and the interaction modeling.

**Keywords** knowledge graph; representation learning; specific domain knowledge graph; interaction among triple elements; link prediction; triplets classification

## 1 引 言

随着大数据时代的到来,现实世界中不同对象(实体或概念)之间的联系日益复杂,相应产生的数据量更是以指数形式增长,呈现出大规模数据关联、交叉和融合的局面[1-2].根据文献[3],大数据的内涵已不再局限于数据,还包括知识以及两者的复合体.因此,使用传统方式对当今世界的数据进行建模已经不再现实.

于是,一系列新技术手段应运而生.知识图谱便是其中之一.它提出的目标是描述真实世界中的实体或概念及其之间的关系.与传统方式相比,其提供了一种新的海量数据组织、管理和利用方式.目前,知识图谱已经在知识问答[4-6]、智能搜索[7]、推荐系统[8]和机器翻译[9]等领域扮演着重要角色.

由于建模基础为现实世界,所以知识图谱多以文字等符号化形式表示.但是,要实现其价值的进一步挖掘,就需要充分利用现有以机器学习为代表的数据挖掘技术.而机器学习模型的输入输出往往都是数值化向量.因此,需要将原有符号化知识图谱转换为数值化表示形式,从而实现量化计算.这里的数值化表示正是知识图谱表示学习的结果.因此,知识图谱表示学习就是通过学习将已有知识图谱的符号化表示转换为数值化表示,从而奠定知识图谱量化计算的基础.

依据不同标准,可以将知识图谱划分为不同类型.从所涉领域的角度出发,可以将其分为通用领域(open domain)和特定领域(specific domain).其中,前者以百科数据集为典型代表,具体包括Freebase[10]、YAGO[11]、WikiData[12]、DBPedia[13]、Nell[14]、Probase[15]和谷歌的 Knowledge Vault[7]

等；而后者则以科学研究、电子商务和医疗等具体领域为主，如生命科学领域的 Bio2RDF[16] 和 Gene Ontology[17]，以及描述家谱关系的 Kinship[18] 等.

经过近十年发展，知识图谱表示学习领域涌现了很多模型. 现有模型多集中于通用领域，而相应验证模型的实验数据集基本都是 WN18[19] 和 FB15k[19] 等通用领域知识图谱的数据样本. 一方面，与通用领域相比，特定领域的语义关系更为集中，相应的数据分布也存在很大不同，尤其是在特定应用前提下. 另一方面，现有模型多从相应数据的具体特征出发构建而成，导致模型的应用过于依赖数据集本身的特征，从而局限了相关技术在现实生产实践中的推广. 因此，如何进行特定领域的知识图谱表示学习就成为一个新问题.

下面以微生物领域的栖息地知识图谱和电子商务领域的用户消费行为知识图谱[①]为例，从拓扑结构和数据分布两方面说明特定领域知识图谱的特殊性. 微生物领域的栖息地知识图谱描述了微生物的栖息地信息. 因此，其有微生物和栖息地两类实体，以及“live_in”关系. 而电子商务领域的用户消费行为知识图谱则介绍了部分用户从 2016 年 2 月到 4 月的点击、浏览、关注、加购（加入购物车）、删购（从购物车删除）和下单六种行为. 所以，其包含六种关系，以及对应的用户和商品两类实体. 两个知识图谱的基本统计信息如表 1 所示.

**表 1　特定领域知识图谱统计信息**

| 数据集 | ＃关系 | ＃实体 | ＃三元组 |
|---|---|---|---|
| 栖息地知识图谱 | 1 | 147 705 | 413 750 |
| 用户消费行为知识图谱 | 6 | 133 890 | 7 992 790 |

（1）拓扑结构. 由于只包含微生物和栖息地之间的“live_in”关系，因此，栖息地知识图谱的头实体均为微生物，尾实体均为栖息地. 虽然只有一种关系，但由于同一个微生物可以栖息在不同地方，而同一环境也可以为不同微生物提供住所，所以栖息地知识图谱也包括形状各异的子图结构. 不过，由于“live_in”关系不具有自反性以及微生物类别和栖息地类别的绝对不交叉特征，所以栖息地知识图谱表现出如图 1 所示的“类二部图结构”. “二部图”的概念建立在无向图基础上，即图中所有节点可以被划分为两个互不交叉的子集，且由每条边连接的两个节点分别属于这两个子集. 但是，由于语义描述需要，知识图谱中的关系往往是从头实体指向尾实体的有向箭头，即知识图谱一般被看作有向图. 故基于“二部图”的特征，本文定义如下“类二部图结构”概念.

**定义 1.**　类二部图结构. 给定知识图谱 $KG=(H,R,T)$，$H$、$R$ 和 $T$ 分别为头实体、关系和尾实体集合. 若 $H\cap T=\varnothing$，则称该知识图谱具有“类二部图结构”.

电子商务领域的用户消费行为知识图谱包括点击、浏览、关注、加购、删购和下单共 6 种关系，以及头实体用户和尾实体商品. 由于头尾实体之间互不交叉，且每种消费行为（即关系）都发生在两者之间，所以，如图 2 所示，用户消费行为知识图谱也具有“类二部图”结构. 但是，从图 1 和图 2 可以看出，栖息地知识图谱只有“live_in”关系，而用户消费行为知识图谱则包括 6 种关系.

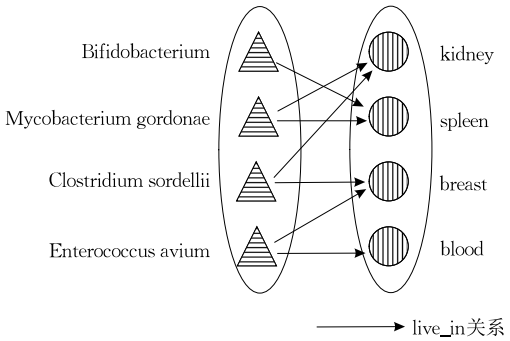


图 1　栖息地知识图谱的“类二部图结构”

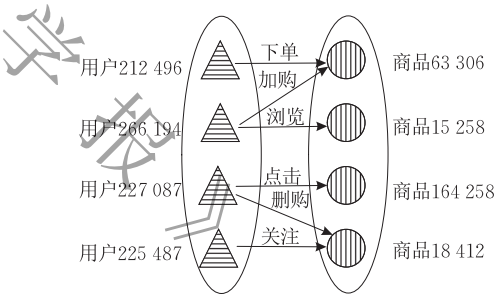


图 2　用户消费行为知识图谱的“类二部图结构”

（2）数据分布. 栖息地知识图谱的头实体均为微生物，尾实体均为栖息地. 图 3 是头尾实体分布直方图. 其中，横轴表示频次分组情况，左侧纵轴表示该频次分组对应的头/尾实体数目，右侧纵轴则表示该频次分组对应的头/尾实体累积频率. 由图可知，栖息地知识图谱的实体分布具有典型的“长尾现象”，尤其是头实体. 比如，出现 10 次以内的头实体共有 134 318 个，占所有头实体的 97％以上；而相应尾实体共有 13 310 个，占比也达到了 90％以上. 除此之外，头实体的最高出现频次是 2316 次，而尾实

① 数据集来自于京东 jdata 算法大赛-高潜用户购买意向预测，网址为 https://www.datafountain.cn/competitions/247/details/data-evaluation.

体却可以达到 27 810 次. 因此,尾实体的平均出现次数更多,即其分布更为集中.

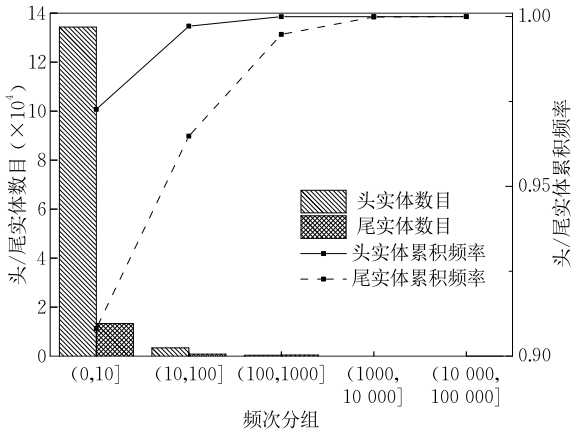


图 3 栖息地知识图谱实体分布直方图

那电子商务领域的用户消费行为知识图谱是否也存在类似特征呢? 图 4 是其实体分布直方图. 与栖息地知识图谱相比,主要有两点不同:(1)从整体分布来看,用户消费行为知识图谱的头尾实体频次更为分散. 在栖息地知识图谱中,出现频率较低的头/尾实体更多,尤其是头实体,频次范围为(0,10]的头实体已经达到了将近 140 000;而用户消费行为知识图谱中相应头实体仅有 70 000 多. 同时,用户消费行为知识图谱的高频次头尾实体较多,尤其是(1000,10 000]和(10 000,100 000]范围的尾实体. (2)用户消费行为知识图谱中的实体更稠密. 其中,出现频次为(0,10]的头实体占比 70%左右,尾实体占比仅有 40%左右,远低于栖息地知识图谱中的 97% 和 90%. 而出现频次为(1000,10 000]和(10 000,100 000]范围的尾实体占比 6%左右,高于栖息地知识图谱中的 0.5%. 因此,用户消费行为知识图谱并不具有典型的"长尾现象". 但两者也有相同点:(1)尾实体均比头实体少;(2)在低频次范围内,头实体占比都比尾实体高.

经过上述分析,可以发现:(1)特定领域知识图谱的数据特征不同于通用领域;(2)就特定领域知识图谱而言,不同领域各有特色. 本文的目标是解决领域知识图谱表示学习问题. 那么其它领域的知识图谱又呈现何种数据分布呢? 我们应该如何抽取不同领域知识图谱的共性来完成特定领域知识图谱表示学习建模呢?

由于穷尽所有领域知识图谱的数据特征并不现实,本文将从比数据分布更抽象的角度着手解决该问题,即从知识图谱构建语义联系的本质特征出发,以三元组为单位,基于头尾实体和关系之间的三元交互提出 InterTris (Interaction among Triple elements)模型. 本文的具体组织结构如下:第 2 节分析已有工作;第 3 节基于理论分析进行模型构建;第 4 节将通过实验验证模型有效性;第 5 节总结全文,并进行展望.

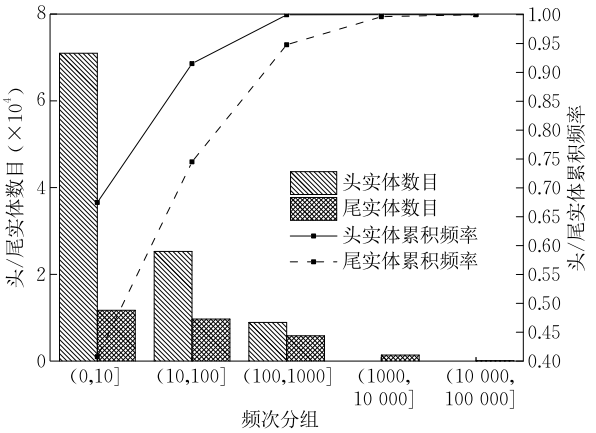


图 4 用户消费行为知识图谱实体分布直方图

## 2 相关工作

已有知识图谱表示学习模型可分为转换(translation)模型①、组合(composition)模型和神经网络(neural network)模型. 下面将分别进行介绍. 本文使用 $h$、$r$ 和 $t$ 表示头实体、关系和尾实体;加粗之后为对应列向量;大写加粗的字母是矩阵. 其它符号将在使用时进行说明.

### 2.1 转换模型

从训练方式来看,可以将 SE[20] 划分为神经网络模型. 但是,如果以建模思路为分类标准,SE 则是转换模型的基础. 通过分别为头实体 $\boldsymbol{h}$ 和尾实体 $\boldsymbol{t}$ 构建相应的映射矩阵 $\boldsymbol{M}_h$ 和 $\boldsymbol{M}_t$,基于相似性度量公式 $S(\boldsymbol{h},\boldsymbol{t})=\|\boldsymbol{M}_h\boldsymbol{h}-\boldsymbol{M}_t\boldsymbol{t}\|_p$,SE 认为相似性越低,构成三元组的概率就越高. 虽然 SE 对关系和头尾实体之间的联系分别进行了建模,但是由于将所有关系同质化,其无法体现知识图谱本体层常见的层次关系.

作为首个最经典的转换模型,TransE[19] 将关系建模为转换操作,认为向量化的头实体 $\boldsymbol{h}$、关系 $\boldsymbol{r}$ 和尾实体 $\boldsymbol{t}$ 满足 $\boldsymbol{h}+\boldsymbol{r}\approx\boldsymbol{t}$. 与其它模型相比,它使用 3 个低维向量完成了对头尾实体和关系的建模. 这

① 也有学者将 translation 模型称为"翻译模型". 但本文认为,模型在构建过程中将头实体(或关系或尾实体)的表示进行了某种程度的"转换",与中文"翻译"的含义略有不同. 故,将 translation 模型称为"转换模型".

种简单性所带来的优势在模型训练中表现十分明显.但是,由于未考虑关系的具体属性,TransE 仅适用于 1-1 和非自反(irreflexive)关系.

为进一步区分不同关系,TransH[21]以关系为单位,首先通过向量 $\boldsymbol{r}_p$ 确定关系所对应的超平面;然后,在超平面上定义转换操作 $\boldsymbol{d}_r$;最后,基于超平面得到映射后头实体向量 $\boldsymbol{h}_\perp=\boldsymbol{h}-\boldsymbol{r}_p^{\mathrm{T}}\boldsymbol{h}\boldsymbol{r}_p$ 和映射后尾实体向量 $\boldsymbol{t}_\perp=\boldsymbol{t}-\boldsymbol{r}_p^{\mathrm{T}}\boldsymbol{t}\boldsymbol{r}_p$,并通过满足约束 $\boldsymbol{h}_\perp+\boldsymbol{d}_r\approx\boldsymbol{t}_\perp$ 进行训练.因此,TransH 通过引入基于关系的超平面,实现了相同实体在不同关系超平面上的不同表示,合理拟合了同一实体在不同关系中的角色变化.

实体往往包括多种含义,而一个关系却只能体现其中一个或几个特征.而且,就实体而言,如果两个实体在某个含义上比较相似,那么它们在语义空间中的距离就应该更近.反之亦然.所以,TransR[22]认为 TransE 和 TransH 的缺陷在于将实体和关系映射到相同空间.为此,其分别定义了实体和关系空间,并在相应空间中构建了实体和关系向量.具体地,首先分别通过 $\boldsymbol{M}_r\boldsymbol{h}$ 和 $\boldsymbol{M}_r\boldsymbol{t}$ 将头尾实体从实体空间映射到关系空间中得到 $\boldsymbol{h}_r$ 和 $\boldsymbol{t}_r$,然后基于关系向量 $\boldsymbol{r}$ 完成转换操作,即满足 $\boldsymbol{h}_r+\boldsymbol{r}\approx\boldsymbol{t}_r$.

TransD[23]和 TranSparse[24]均为 TransR 的变体.与 TransR 考虑关系多样性不同,TransD 的目标在于解决实体多样性问题.其中,每个头尾实体和关系都有两个向量:一个是语义向量 $\boldsymbol{h}$、$\boldsymbol{r}$、$\boldsymbol{t}$;另一个则是用于动态构建映射矩阵 $\boldsymbol{M}_r^h=\boldsymbol{r}_p\boldsymbol{h}_p^{\mathrm{T}}+\boldsymbol{I}^{m\times n}$ 和 $\boldsymbol{M}_r^t=\boldsymbol{r}_p\boldsymbol{t}_p^{\mathrm{T}}+\boldsymbol{I}^{m\times n}$ 的映射向量 $\boldsymbol{h}_p$、$\boldsymbol{r}_p$ 及 $\boldsymbol{t}_p$.由于在具体构建映射矩阵过程中,同时考虑了实体和关系的影响,所以,TransD 实现了实体-关系交互建模.同时,与 TransR 相比,TransD 中只有向量乘,不存在矩阵乘,所以复杂度更低.

与已有工作不同,TranSparse[24]发现了知识图谱中普遍存在的关系异质性和不平衡问题.其中,前者指不同关系连接的头尾实体数目差距较大;而后者则指相同关系连接的头实体数目和尾实体数目不同.其认为,关系的语义复杂度与其连接的头尾实体数相关;而语义越复杂,就应该用更多的参数来表示该关系与对应实体之间的交互.具体地,则通过将 TransR 中的普通稠密矩阵替换为稀疏矩阵完成建模.

因此,转换模型往往将头尾实体和关系建模为矩阵或向量,时间和空间复杂度较低.但是,其一般都使用向量或者矩阵乘完成交互建模,所以表达能力有限.

### 2.2　组合模型

由于将知识图谱建模为三维邻接张量,组合模型又被称为"张量分解"(tensor factorization)模型.RESCAL[25]是最经典的组合模型,其使用三维邻接张量 $\boldsymbol{\chi}$ 表示三元组,并通过张量分解进行建模.而 LFM[26]则将头尾实体建模为 $\boldsymbol{h}$、$\boldsymbol{t}$,将关系建模为 $\boldsymbol{M}_r$;基于自然语言处理中的一元(unigram)、二元(bigram)和三元(trigram)现象,在实体嵌入式表示过程中引入了二阶相关性;并根据目标函数 $\boldsymbol{h}^{\mathrm{T}}\boldsymbol{M}_r\boldsymbol{t}$ 进行模型调优.因此,它以一种相对简单有效的方式实现了实体之间的交互建模.

基于 TransE[19]和 NTN[27]等已有嵌入式表示学习模型,DistMult[28]提出了可以将这些模型统一起来的框架 $\boldsymbol{y}_h^{\mathrm{T}}\boldsymbol{M}_r\boldsymbol{y}_t$($\boldsymbol{y}_h$ 和 $\boldsymbol{y}_t$ 是头尾实体的向量化表示函数,$\boldsymbol{M}_r$ 为关系的矩阵表示).为降低复杂度,模型将 $\boldsymbol{M}_r$ 定义为对角矩阵,具有和 TransE[19]相同的参数规模.

HolE[29]主要通过循环相关对实体之间的丰富交互进行了建模,即 $[\boldsymbol{h}\times\boldsymbol{t}]_k=\sum_{i=0}^{n-1}\boldsymbol{h}\boldsymbol{t}_{(i+k)\bmod n}$.Complex[30]在其基础之上,将所有实数向量转换为复数向量进行建模求解.ANALOGY[31]则通过考虑相似属性,综合了 DistMult、HolE 和 Complex 共 3 个模型的优势.

因此,组合模型的建模基础从张量分解发展到向量内积,在提高表达力的同时,降低了时间和空间复杂度.但是,其在交互建模过程中,考虑的多为头实体-尾实体或实体-关系之间的交互,建模并不是很充分.

### 2.3　神经网络模型

作为较早的神经网络模型,SME[32]将头尾实体和关系表示为向量,基于映射矩阵,通过矩阵向量乘和 Hadamard 积实现了实体-关系交互建模.对应目标函数有两种定义,即线性目标函数 $f_r(\boldsymbol{h},\boldsymbol{t})=(\boldsymbol{M}_1\boldsymbol{l}_h+\boldsymbol{M}_2\boldsymbol{l}_r+\boldsymbol{b}_1)^{\mathrm{T}}(\boldsymbol{M}_3\boldsymbol{l}_t+\boldsymbol{M}_4\boldsymbol{l}_r+\boldsymbol{b}_2)$ 和双线性目标函数 $f_r(\boldsymbol{h},\boldsymbol{t})=(\boldsymbol{M}_1\boldsymbol{l}_h\otimes\boldsymbol{M}_2\boldsymbol{l}_r+\boldsymbol{b}_1)^{\mathrm{T}}(\boldsymbol{M}_3\boldsymbol{l}_t\otimes\boldsymbol{M}_4\boldsymbol{l}_r+\boldsymbol{b}_2)$.其中,$\otimes$ 为 Hadamard 积.

与 SE 相比,神经网络模型 SLM[27]通过非线性计算对实体-关系之间的语义联系进行了建模.具体而言,其将头尾实体 $\boldsymbol{h}$ 和 $\boldsymbol{t}$ 作为神经网络模型隐藏层的输入,由输出层按照目标函数 $f_r(\boldsymbol{h},\boldsymbol{t})=\boldsymbol{u}_r^{\mathrm{T}}g(\boldsymbol{M}_{r,1}\boldsymbol{l}_h+\boldsymbol{M}_{r,2}\boldsymbol{l}_t)$ 计算得分.

基于 SLM,NTN[27]在神经网络的非线性计算中考虑了二阶相关性,从多个维度出发将相应头尾

实体的向量联系起来，从而根据目标函数 $f_r(\boldsymbol{h},\boldsymbol{t})=\boldsymbol{u}_r^{\mathrm{T}}g(\boldsymbol{l}_h\boldsymbol{M}_r\boldsymbol{l}_t+\boldsymbol{M}_{r,1}\boldsymbol{l}_h+\boldsymbol{M}_{r,2}\boldsymbol{l}_t+\boldsymbol{b}_r)$ 得到对应三元组成立的概率. 虽然 NTN 可以囊括众多表示学习模型，表达能力也很强，但其复杂度往往难以满足现实需求.

因此，神经网络模型在发展中越来越复杂，虽然提高了表达能力，但却往往难以投入实践.

综合以上三类模型，一方面，从发展历程来看，它们都在不断提升数据描述能力. 由于面向数据特征进行建模，而不同知识图谱的数据分布又有很大差异，尤其是通用领域知识图谱和特定领域知识图谱. 所以，表示学习模型在不同数据集上的表现并不稳定. 而且，三类模型目前均面向通用领域构建，在应用到特定领域时会面临新的数据分布挑战. 因此，有必要针对特定领域构建相应的知识图谱表示学习模型. 另一方面，现有模型尚未对头实体、尾实体和关系三者之间的丰富交互进行充分描述. 在三类模型中，只有一个充分建模了头尾实体和关系之间的交互作用，即 NTN. 但是，NTN 的复杂度却成为现实应用的最大障碍. 因此如何在建模充分性和复杂度之间达到较好的平衡依旧是个需要研究的问题.

## 3 InterTris 模型

与通用领域相比，特定领域的数据分布更为集中，从而产生“长尾”等特征. 如引言部分所述，特定领域知识图谱的数据特征不仅与通用领域不同，不同领域的知识图谱之间也有差异，而且这种差异无法穷尽. 因此，从数据特征的角度出发进行特定领域的知识图谱表示学习建模并非最佳选择. 那什么才是更合适的建模角度呢？

### 3.1 模型构建

基于前述分析，可以发现，要为特定领域知识图谱构建统一的表示学习模型，就必须找到它们之间的共性. 所以，本文从知识图谱建模现实世界的方式等角度出发进行模型构建.

与其它很多新生技术一样，知识图谱的诞生也有其现实驱动力，即在新生事物不断涌现，已有事物和新事物之间的联系不断丰富的时代背景下，如何对现有数字世界的信息进行重组，从而更好地理解现实世界. 知识图谱建模现实世界的方式是将实体或概念表示为节点，将它们之间的关系表示为边，从而形成一张巨大的语义网. 除此之外，知识图谱也可表示为“属性图”的形式. 但是，如果将属性看作一种特殊的关系，则属性值可以表示为尾实体. 因此，可以认为知识图谱的本质是构建语义联系，而三元组就是其基本单位. 所以与一般数据集相比，三元组是知识图谱特有的结构；同时，也是所有特定领域知识图谱，都具有的结构. 因此如果从三元组粒度出发进行建模，模型将适用于不同的领域知识图谱.

下面将围绕三元组中头实体、关系和尾实体三者本身及其之间的交互进行建模. 首先，需要为三者构建语义向量，即 $\boldsymbol{h},\boldsymbol{r},\boldsymbol{t}\in\boldsymbol{R}^n$. 其次，为了充分描述头尾实体和关系之间的交互，本文认为三者中的任何一个都会受到另外两个因素的影响. 因为现实中会存在以下 3 种情况：(1) 相同头实体和关系对应不同尾实体的例子，如〈美国，总统，奥巴马〉和〈美国，总统，特朗普〉；(2) 相同头实体和尾实体对应不同关系的例子，如〈乔布斯，创始人，苹果公司〉和〈乔布斯，CEO，苹果公司〉；(3) 相同关系和尾实体对应不同头实体的例子，如〈奥巴马，国籍，美国〉和〈特朗普，国籍，美国〉. 这里用向量 $\boldsymbol{h}_p,\boldsymbol{r}_p,\boldsymbol{t}_p\in\boldsymbol{R}^n$ 分别表示头实体、关系和尾实体对另外两个元素的影响. 最后，进行交互过程建模. 目前已有向量相乘和向量内积两种建模方式. 前者得到的低秩矩阵会局限模型表达能力；而向量内积却是向量中每个元素之间的相乘，可以实现元素级别的建模. 所以，本文采用第 2 种方式.

如图 5 所示，以头实体为例，其语义向量为 $\boldsymbol{h}$，由于映射后头实体 $h_\perp\in\boldsymbol{R}$ 会同时受到其本身，以及关系和尾实体的影响. 基于广义内积对交互过程建模，可以得到 $h_\perp$，即

$$h_\perp=\langle\boldsymbol{h},\boldsymbol{r}_p,\boldsymbol{t}_p\rangle \tag{1}$$

其中，映射向量 $\boldsymbol{r}_p$ 和 $\boldsymbol{t}_p$ 分别表示关系和尾实体对头实体的影响；$\langle\boldsymbol{h},\boldsymbol{r}_p,\boldsymbol{t}_p\rangle$ 表示 $\boldsymbol{h}$、$\boldsymbol{r}_p$ 和 $\boldsymbol{t}_p$ 的广义内积. 同理，可得到映射后的关系和尾实体

$$r_\perp=\langle\boldsymbol{h}_p,\boldsymbol{r},\boldsymbol{t}_p\rangle;\ t_\perp=\langle\boldsymbol{h}_p,\boldsymbol{r}_p,\boldsymbol{t}\rangle \tag{2}$$

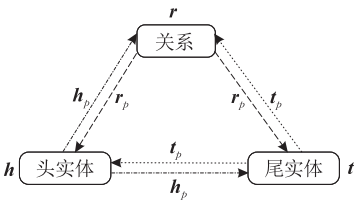


图 5 InterTris 示例

最后，基于转换思想，模型的目标函数定义为

$$f(\boldsymbol{h},\boldsymbol{r},\boldsymbol{t})=h_\perp+r_\perp-t_\perp \tag{3}$$

为了使模型尽快收敛，训练过程中需增加的约束包括 $\|\boldsymbol{h}\|\leq1$，$\|\boldsymbol{r}\|\leq1$，$\|\boldsymbol{t}\|\leq1$，$\|\boldsymbol{h}_p\|\leq1$，$\|\boldsymbol{r}_p\|\leq1$ 和 $\|\boldsymbol{t}_p\|\leq1$.

综上，为了解决不同领域的数据分布差异所带来的知识图谱表示学习建模挑战，本文从更为抽象

的知识图谱建模语义联系的角度出发，以三元组为基本单位.同时，考虑到现实世界中头实体、关系和尾实体三者中的任意一个都会同时受到另外两个元素的影响，我们形成了三元交互的基本建模思路，从而提出基于三元交互的知识图谱表示学习模型InterTris.

### 3.2　模型训练

基于已经建立的InterTris模型，本节将对训练相关细节进行阐述.一方面，主要介绍模型的训练方式，包括损失函数、参数更新所采用的优化算法等.另一方面，依据所选择的训练方式，以算法形式详细介绍具体训练过程.

假设训练集 $S$ 包括 $s$ 个三元组，第 $i$ 个三元组是 $\langle h_i,r_i,t_i\rangle(i=1,\cdots,s)$.每个三元组都有标签 $y_i(i=1,\cdots,s)$.若 $y_i=1$，相应三元组成立；$y_i=0$，则三元组不成立.所有成立的三元组构成正例集合 $\Delta$，不成立的构成负例集合 $\Delta'$.但是，由于本文数据集只有正例，没有负例，所以这里采用bern[21]方法进行负例生成.之所以采用bern方法，是因为其降低了假负例生成概率.具体采样过程如下：首先，对于给定的关系 $r$ 计算出头实体平均连接的尾实体数目 $tph$，以及尾实体平均连接的头实体数目 $hpt$；然后，以 $tph/(tph+hpt)$ 为参数定义伯努利采样，即生成负例时，使用其它实体替换头实体的概率是 $tph/(tph+hpt)$，替换尾实体的概率是 $hpt/(tph+hpt)$.

与基于margin的ranking loss函数相比，Liu等人[31]发现基于sigmoid的logistic损失函数更好.因此，本文定义如下的sigmoid函数

$$\sigma=\begin{cases}0, & f(\boldsymbol{h},\boldsymbol{r},\boldsymbol{t})<-cutoff\\ 1, & f(\boldsymbol{h},\boldsymbol{r},\boldsymbol{t})>cutoff\\ \dfrac{1}{1+\exp(-f(\boldsymbol{h},\boldsymbol{r},\boldsymbol{t}))}, & \text{其他}\end{cases}\tag{4}$$

其中，$cutoff\in(0,+\infty)$ 为临界值.由于sigmoid函数的定义域为 $(-\infty,+\infty)$，值域为 $(0,1)$；且自变量越小，取值越接近于0，反之则越接近1.为防止梯度消失，同时也为了降低计算成本，本文假设了临界值 $cutoff$，认为当 $f(\boldsymbol{h},\boldsymbol{r},\boldsymbol{t})$ 大于 $cutoff$ 时，则经过sigmoid函数计算之后的值为1，若小于 $-cutoff$，则计算值为0.因此，原有的连续sigmoid函数成为式(4)所示的分段函数.基于此，本文的损失函数为

$$\mathcal{L}=\min\sum_{\langle\boldsymbol{h},\boldsymbol{r},\boldsymbol{t}\rangle\in S}(y_i(\log\sigma(f(\boldsymbol{h},\boldsymbol{r},\boldsymbol{t})))+(1-y_i)\log(1-\sigma(f(\boldsymbol{h},\boldsymbol{r},\boldsymbol{t}))))\tag{5}$$

为避免参数初始化影响实验结果，本文在实验中，对所有向量进行随机初始化，同时使用单位矩阵初始化所有矩阵.考虑到生产实践的效率需求，InterTris采用了基于AdaGrad[33]，即改进版的mini-batch SGD优化算法.虽然SGD和AdaGrad都能实现最终收敛，但是后者的收敛速度更快.同时，本文的实验也采用了基于Hogwild![34]的并行优化算法框架，并行线程数为32.虽然并行处理会影响最终实验结果，但误差在可接受的范围之内.无论是AdaGrad还是Hogwild!，这两个训练设置都是为了加速收敛过程，降低计算成本.

如算法1所示，InterTris的训练过程主要包括三部分：第1行是输入部分；第2～13行是参数初始化部分；第14～29行则是模型迭代部分.在模型迭代部分，算法首先通过随机抽样得到对应的mini-batch数据集；然后基于bern采样方法得到相应负例；接着便依据式(5)中的损失函数，进行参数更新；为使模型更快收敛，需要在参数更新完成后，对语义和映射向量进行正则化，即第21～28行.

**算法1.**　InterTris训练过程.

1. 输入：训练数据集 $\Delta=\{\langle\boldsymbol{h},\boldsymbol{r},\boldsymbol{t}\rangle\}$；头实体集合 $H$；关系集合 $R$；尾实体集合 $T$；临界值 $cutoff$；头实体、关系和尾实体的语义向量和映射向量维度均为 $n$.
2. 初始化 $\boldsymbol{h}\leftarrow uniform\left(-\frac{1}{n},\frac{1}{n}\right)$ FOR EACH $h\in H$
3. $\boldsymbol{r}\leftarrow uniform\left(-\frac{1}{n},\frac{1}{n}\right)$ FOR EACH $r\in R$
4. $\boldsymbol{t}\leftarrow uniform\left(-\frac{1}{n},\frac{1}{n}\right)$ FOR EACH $t\in T$
5. $\boldsymbol{h}_p\leftarrow uniform\left(-\frac{1}{n},\frac{1}{n}\right)$ FOR EACH $h\in H$
6. $\boldsymbol{r}_p\leftarrow uniform\left(-\frac{1}{n},\frac{1}{n}\right)$ FOR EACH $r\in R$
7. $\boldsymbol{t}_p\leftarrow uniform\left(-\frac{1}{n},\frac{1}{n}\right)$ FOR EACH $t\in T$
8. $\boldsymbol{h}\leftarrow\boldsymbol{h}/\|\boldsymbol{h}\|$ FOR EACH $h\in H$
9. $\boldsymbol{r}\leftarrow\boldsymbol{r}/\|\boldsymbol{r}\|$ FOR EACH $r\in R$
10. $\boldsymbol{t}\leftarrow\boldsymbol{t}/\|\boldsymbol{t}\|$ FOR EACH $t\in T$
11. $\boldsymbol{h}_p\leftarrow\boldsymbol{h}_p/\|\boldsymbol{h}_p\|$ FOR EACH $h\in H$
12. $\boldsymbol{r}_p\leftarrow\boldsymbol{r}_p/\|\boldsymbol{r}_p\|$ FOR EACH $r\in R$
13. $\boldsymbol{t}_p\leftarrow\boldsymbol{t}_p/\|\boldsymbol{t}_p\|$ FOR EACH $t\in T$
14. LOOP
15. $\Delta_{\text{batch}}\leftarrow$sample$(\Delta,b)$ //随机抽样出包含 $b$ 个三元组的样本
16. $T_{\text{batch}}\leftarrow0$//初始化三元组的正负例对集合
17. FOR $\langle h,r,t\rangle\in\Delta_{\text{batch}}$ DO
18. $\langle h',r,t'\rangle\leftarrow sample(\Delta'_{\langle h,r,t\rangle})$ //基于bern方法抽样生成负例
19. END FOR
20. 基于损失函数更新语义向量和映射向量，即

$$\min \sum_{\langle h,r,t\rangle \in S} (y_i(\log\sigma(f(\boldsymbol{h},\boldsymbol{r},\boldsymbol{t}))) + (1-y_i)\cdot \log(1-\sigma(f(\boldsymbol{h},\boldsymbol{r},\boldsymbol{t}))))$$

21. FOR $\ell \in H,R,T$ in $T_{\text{batch}}$ DO
//正则化语义和映射向量
22. IF $\|\ell\|>1$ THEN
23. $\ell \leftarrow \ell/\|\ell\|$
//满足 $\|\boldsymbol{h}\|\leq 1;\|\boldsymbol{r}\|\leq 1;\|\boldsymbol{t}\|\leq 1$
24. END IF
25. IF $\|\ell_p\|>1$ THEN
26. $\ell_p \leftarrow \ell_p/\|\ell_p\|$
//满足约束 $\|\boldsymbol{h}_p\|\leq 1;\|\boldsymbol{r}_p\|\leq 1;\|\boldsymbol{t}_p\|\leq 1$
27. END IF
28. END FOR
29. END LOOP

### 3.3 复杂度分析

由于生产实践过程对模型效率要求较高，所以，本小节将基于已有模型，对 InterTris 模型的复杂度进行分析. 这里的时间复杂度指一轮训练过程所需计算的乘法次数；而空间复杂度则指参数数量. 相关符号的定义如下：$N$、$N_r$ 和 $N_e$ 分别指训练集中三元组、关系和实体的数目；$m$ 是实体的向量维数，而 $n$ 则是关系的向量维数；在组合模型中，$s$ 表示张量的切片数；在神经网络模型中，$k$ 表示隐藏层的节点数；对于转换模型中的 TranSparse 而言，$\theta_{\text{avg}}$ 表示所有映射矩阵(包括头实体和尾实体对应的两类映射矩阵)稀疏度的平均值. 各模型的时间和空间复杂度详见表 2.

**表 2 各模型复杂度对比分析**

| 模型 | 时间复杂度 | 空间复杂度 |
|---|---|---|
| TransE[19] | $O(N)$ | $O(N_em+N_rn)(m=n)$ |
| TransH[21] | $O(2mN)$ | $O(N_em+2N_rn)(m=n)$ |
| TransR[22] | $O(2mnN)$ | $O(N_em+N_r(m+1)n)$ |
| TransD[23] | $O(2nN)$ | $O(2N_em+2N_rn)$ |
| TranSparse[24] | $O(2(1-\theta_{\text{avg}})mnN)$ $(0\ll\theta_{\text{avg}}\leq 1)$ | $O(N_em+2N_r(1-\theta_{\text{avg}})(m+1)n)$ $(0\ll\theta_{\text{avg}}\leq 1)$ |
| DistMult[28] | $O(2mN)$ | $O(N_em+N_rn^2)(m=n)$ |
| HolE[29] | $O(m\log(m)N)$ | $O(N_em+N_rn)(m=n)$ |
| Complex[30] | $O(4mN)$ | $O(N_em+N_rn)(m=n)$ |
| ANALOGY[31] | $O(3mN)$ | $O(N_em+N_rn)(m=n)$ |
| $\text{SME}_{\text{linear}}$[32] | $O(4mkN)$ | $O(N_em+N_rn+4mk+4k)(m=n)$ |
| $\text{SME}_{\text{bilinear}}$[32] | $O(4mksN)$ | $O(N_em+N_rn+4mks+4k)(m=n)$ |
| SLM[27] | $O((2mk+k)N)$ | $O(N_em+N_r(2k+2nk))(m=n)$ |
| NTN[27] | $O(((m^2+m)s+2mk+k)N)$ | $O(N_em+N_r(n^2s+2ns+2s)(m=n)$ |
| InterTris | $O(2nN)$ | $O(2N_em+2N_rn)(m=n)$ |

由表 2 可知，相对于其它两类模型，转换模型的时间和空间复杂度较低. 其中，TransR 和 TranSparse 略高，而 TransE、TransH 和 TransD 这几个模型在所有对比模型中都属于比较简单的，尤其是 TransE. 组合模型的时间复杂度按照 DistMult、HolE、Complex 和 ANALOGY 的顺序，先增加后减少，但 ANALOGY 的时间复杂度依旧高于 DistMult；从空间复杂度来看，这类模型可以分为 DistMult 和其它模型两类，前者为 $O(N_em+N_rn^2)(m=n)$，比后者的 $O(N_em+N_rn)(m=n)$ 更高. 对于神经网络模型而言，无论是时间复杂度还是空间复杂度，它们都按照从 $\text{SME}_{\text{linear}}$、$\text{SME}_{\text{bilinear}}$、SLM 再到 NTN 的顺序，越来越高. 本文的 InterTris 模型在时间和空间复杂度上与 TransD 相当. 因此，与已有模型相比，InterTris 在复杂度方面的表现较好.

## 4 实验结果及其分析

与 NTN[27] 类似，几乎所有神经网络模型都需要学习很多参数，从而带来较高计算成本，无法满足实践需求，所以这里不将其纳入对比范围. 下面将以较优的 5 个转换模型和 4 个组合模型作为基线(baseline)产生进行对比实验. 其中，前者包括 TransE[19]、TransH[21]、TransR[22]、TransD[23] 和 TranSparse[24]；后者包括 DistMult[28]、HolE[29]、Complex[30] 和 ANALOGY[31]. 具体的实验任务包括链接预测[19] 和三元组分类[27].

### 4.1 实验数据

如前所述，基于三元组粒度，InterTris 对头实体、关系和尾实体三者之间的交互进行了充分建模，可以更好解决特定领域的知识图谱表示学习问题. 为了检验 InterTris 在不同领域知识图谱中的应用效果，这部分实验数据集包括家谱领域的公共知识图谱 Kinship[18]、微生物领域的酶知识图谱样本 ES[35]、微生物领域的栖息地知识图谱样本 LiveIn 和电子商务领域的用户消费行为知识图谱样本 UserAct.

Bordes[19] 在构建 FB15k 和 FB1M 时，选择了高频实体所在三元组. 虽然这样可使模型得到更好的训练，但却无法充分体现原有数据集的整体分布. 为了避免这种问题，本文基于简单随机抽样构建了微生物领域的栖息地知识图谱样本 LiveIn 和电子商务领域的用户消费行为知识图谱样本 UserAct，即首先去除仅出现过一次的实体及其对应三元组；然后进行简单随机抽样；最后去除实验数据集中仅出现过一次的实体及其对应三元组. Kinship、ES、LiveIn 和 UserAct 的具体统计信息如表 3 所示.

**表 3　实验数据统计信息**

| 数据集 | #关系 | #实体 | #训练集 | #验证集 | #测试集 |
|---|---|---|---|---|---|
| Kinship | 25 | 104 | 6411 | 2137 | 2138 |
| ES | 10 | 57066 | 155417 | 5000 | 5000 |
| LiveIn | 1 | 45877 | 150000 | 5120 | 5144 |
| UserAct | 6 | 4233 | 144843 | 6000 | 6000 |

### 4.2　链接预测

链接预测指已知头(尾)实体和关系，预测尾(头)实体.本文也包括关系预测.与已有表示学习模型[19-24]类似，本文的评价指标包括 *raw* 和 *filt* 两种情况下的 *Mean Rank* 和 *Hit@k*(%).首先，替换测试集中三元组的头实体、关系或尾实体得到候选三元组；其次，按照目标函数计算候选三元组得分；最后，基于得分对三元组进行成立可能性的降序排列.链接预测重点关注排序的相对正确，即正确三元组的位置.所有三元组的位置平均值为 *Mean Rank*，前 $k$ 个候选三元组的召回率为 *Hit@k*(%).但是实验数据集包括训练集、验证集和测试集.如果候选三元组中包含了这三个数据集中出现过的三元组就会导致正确三元组的排序靠后.所以，使用 *raw* 和 *filt* 来区别过滤前后的两种情况.

由于 Kinship、ES、LiveIn 和 UserAct 中的关系数分别是 25、10、1 和 6，所以，除了 LiveIn 不进行关系预测之外，Kinship 关系预测的 *Hit@k* 中 $k$ 为 10，ES 和 UserAct 关系预测的 *Hit@k* 中 $k$ 为 1.

文献[35]的实验数据集是 Kinship 和 ES，且在实验过程中，将所有模型的嵌入式表示维度都设为 20.为防止维度不同带来偏差，InterTris 也在 20 维基础之上进行调参.基于 AdaGrad，InterTris 的学习率是 0.1，最大训练轮数为 1000.调参范围为学习率衰减系数 $\gamma\in\{0.1,0.01,0.001\}$，负例个数 $n\in\{3,4,5,6\}$.通过网格搜索，本文得到 Kinship 的最优参数 $\gamma=0.001$，$n=5$；ES 的最优参数 $\gamma=0.001$，$n=6$.

LiveIn 和 UserAct 是本文构建的数据集.实验中，基于 AdaGrad 算法，InterTris 学习率为 0.1，最大训练轮数为 1000.调参范围为维度 $k\in\{20,100,150,200\}$，学习率衰减系数 $\gamma\in\{0.1,0.01,0.001,0.0001\}$，负例个数 $n\in\{3,4,5,6\}$.通过网格搜索，得到 LiveIn 最优参数为 $k=200$，$\gamma=0.0001$，$n=4$；UserAct 最优参数为 $k=150$，$\gamma=0.1$，$n=3$.

如前所述，LiveIn 和 UserAct 均通过简单随机抽样从实际数据中抽取得到，受现实条件限制，部分预测的 *Hit@k* 为 0 或接近 0.实验结果中均以“—”表示.此外，对于 4 个数据集而言，加粗结果均表示相应实验条件下的最佳模型.

表 4 是基于家谱领域公共知识图谱 Kinship 的链接预测结果.除了头实体预测 raw 情况下的 *Hit@10* 比 ANALOGY 模型低 1.9 个百分点之外，InterTris 在其它所有指标上的表现均为最优.与 *raw* 相比，*filt* 在获取目标元素(包括头实体、关系和尾实体)排名的时候，删除了其之前所有已知正确的候选对象.而实际应用中并不考虑已知正确的三元组，所以 *filt* 情况下的实验结果更具参考意义，即 InterTris 可以在现实应用中达到最优.

**表 4　基于 Kinship 的链接预测结果**

| 模型 | 头实体预测 | | | | 尾实体预测 | | | | 关系预测 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *Mean Rank* | | *Hit@10* (%) | | *Mean Rank* | | *Hit@10* (%) | | *Mean Rank* | | *Hit@10* (%) | |
| | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* |
| TransE[19] | 23 | 19 | 40.2 | 51.3 | 27 | 21 | 33.6 | 46.0 | 5 | 5 | 84.8 | 84.8 |
| TransH[21] | 20 | 16 | 42.9 | 56.1 | 22 | 17 | 35.5 | 53.3 | 4 | 4 | 91.4 | 91.4 |
| TransR[22] | 14 | 9 | 52.0 | 76.1 | 16 | 10 | 44.7 | 72.1 | 3 | 3 | 94.3 | 94.3 |
| TransD[23] | 15 | 8 | 51.5 | 65.2 | 13 | 12 | 45.9 | 64.4 | 3 | 3 | 92.7 | 92.7 |
| TranSparse[24] | 9 | 5 | 69.4 | 90.2 | 11 | 5 | 57.7 | 88.5 | 2 | 2 | 98.4 | 98.4 |
| DistMult[28] | 16 | 8 | 65.6 | 88.7 | 12 | 6 | 51.6 | 84.6 | 2 | 2 | 98.6 | 98.8 |
| HolE[29] | 14 | 7 | 66.6 | 87.3 | 15 | 7 | 56.3 | 88.6 | 3 | 3 | 97.6 | 97.6 |
| Complex[30] | 11 | 6 | 71.1 | 88.8 | 13 | 6 | 56.3 | 86.3 | 2 | 2 | 98.4 | 98.4 |
| ANALOGY[31] | 10 | 4 | **74.1** | 88.9 | 12 | 5 | 57.4 | 87.4 | 2 | 2 | 98.3 | 98.3 |
| InterTris | **8** | **3** | 72.2 | **95.9** | **10** | **3** | **58.9** | **95.6** | **1** | **1** | **99.1** | **99.4** |

表 5 是基于微生物酶知识图谱样本 ES 的链接预测结果.首先，从整体上来看，进行头实体和关系预测时，*raw* 和 *filt* 的结果十分相近.但是，进行尾实体预测时，*raw Mean Rank* 均高于 600，而 *filt* 之后的结果却可以低到 4.尾实体预测的 *Hit@10* 也有类似情况.具体地，*raw* 情况下的结果均低于 10%，而 *filt* 之后的结果甚至可以达到 90%以上.这是因为 *raw* 和 *filt* 两种情况的区别在于是否从预测结果中删除了训练集、验证集和测试集中已出现的三元组.与头实体和关系预测一样，尾实体预测

也需要通过计算所有候选对象是目标尾实体的可能性并进行排序.但是,ES 的尾实体在实体中占绝大部分.而且相同的头实体和关系可能连接了成千上万个尾实体[35].所以,在预测尾实体时,更有可能删除已出现过的三元组,从而导致了 *raw* 和 *filt* 两种情况下链接预测结果的差异.其次,与其它模型相比,InterTris 在 *filt* 情况下的所有指标均为最优,只有在尾实体和关系的 *raw Hit@k* 设置中稍低于最优结果.但是,由于 *filt* 指标更符合实践情况,所以,其更适合于现实应用.最后,InterTris 的 *filt* 指标较优,*raw* 较差,说明在原始候选对象排序中,目标对象之前的候选对象更多是正确的,证明 InterTris 对已有数据的拟合能力较强.

**表 5 基于 ES 的链接预测结果**

| 模型 | 头实体预测 | | | | 尾实体预测 | | | | 关系预测 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *Mean Rank* | | *Hit@10* (%) | | *Mean Rank* | | *Hit@10* (%) | | *Mean Rank* | | *Hit@1* (%) | |
| | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* |
| TransE[19] | 21 | 18 | 84.7 | 85.9 | 2531 | 1929 | 5.7 | 28.5 | 1 | 1 | 92.1 | 92.1 |
| TransH[21] | 25 | 22 | 86.6 | 87.4 | 1726 | 1114 | 5.4 | 49.6 | 1 | 1 | 93.5 | 93.5 |
| TransR[22] | 26 | 24 | 89.0 | 90.0 | 820 | 202 | 5.2 | 36.8 | 1 | 1 | 98.3 | 98.3 |
| TransD[23] | 38 | 35 | 96.2 | 96.5 | 728 | 137 | 7.3 | 76.1 | 1 | 1 | 92.4 | 93.7 |
| TranSparse[24] | 14 | 11 | 95.2 | 95.7 | 731 | 117 | 7.0 | 50.6 | 1 | 1 | 94.6 | 94.6 |
| DistMult[28] | 52 | 49 | 93.9 | 94.2 | 758 | 130 | 7.8 | 82.3 | 1 | 1 | 98.4 | 98.7 |
| HolE[29] | 17 | 11 | 94.2 | 96.4 | 732 | 124 | **9.0** | 83.0 | 1 | 1 | 99.5 | 99.8 |
| Complex[30] | 29 | 26 | 94.0 | 94.1 | 741 | 128 | 8.6 | 83.9 | 1 | 1 | 99.6 | 99.7 |
| ANALOGY[31] | 16 | 13 | 94.5 | 94.7 | 735 | 126 | 8.9 | 84.8 | 1 | 1 | **99.9** | 99.9 |
| InterTris | **4** | **1** | **97.1** | **99.5** | **607** | **4** | 8.5 | **91.1** | 1 | 1 | 99.8 | **100.0** |

表 6 是基于微生物领域栖息地知识图谱样本 LiveIn 的链接预测结果.首先,InterTris 整体表现最优.其在头实体预测的 *filt Mean Rank* 和 *Hit@10* 以及尾实体预测的 *Mean Rank* 上达到最优.且 *Mean Rank* 值最多可比次优模型低 30.其次,DistMult 在头实体预测的 *raw Mean Rank* 上达到了最优.这是因为 DistMult 使用向量内积对头尾实体和关系三者之间的交互作用进行了建模.这也是其复杂度与 TransE 相当但效果更好的原因之一.之所以没有 InterTris 好,是因为其在建模过程中仅考虑了头尾实体和关系三者之间的交互,并未考虑各元素的语义信息.最后,TransD 在尾实体预测的 *Hit@10* 上达到了最优.这是因为其分别为头实体、关系和尾实体构建了语义和映射向量,且建模了实体-关系交互.但由于并未对头实体、关系和尾实体三者之间的三元交互进行充分建模,所以,TransD在其它指标上逊色于 InterTris.

**表 6 基于 LiveIn 的链接预测结果**

| 模型 | 头实体预测 | | | | 尾实体预测 | | | |
|---|---|---|---|---|---|---|---|---|
| | *Mean Rank* | | *Hit@10* (%) | | *Mean Rank* | | *Hit@10* (%) | |
| | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* |
| TransE[19] | 11536 | 10813 | 0.3 | 0.5 | 10026 | 9991 | 0.3 | 0.3 |
| TransH[21] | 22737 | 22204 | — | — | 22574 | 22542 | — | — |
| TransR[22] | 24989 | 24420 | — | — | 14252 | 14219 | 0.4 | 0.4 |
| TransD[23] | 16818 | 16204 | 0.3 | 0.4 | 898 | 865 | **8.8** | **9.7** |
| TranSparse[24] | 22958 | 22423 | — | — | 21720 | 21688 | — | — |
| DistMult[28] | **8180** | 7216 | 1.5 | 1.6 | 688 | 648 | 6.3 | 7.4 |
| HolE[29] | 8595 | 7627 | 1.6 | 2.1 | 1521 | 1481 | 2.8 | 3.8 |
| Complex[30] | 8336 | 7374 | 1.8 | 2.0 | 947 | 902 | 3.7 | 4.5 |
| ANALOGY[31] | 8442 | 7460 | 1.9 | 2.1 | 941 | 984 | 5.4 | 6.2 |
| InterTris | 8249 | **7210** | **2.6** | **3.4** | **673** | **618** | 5.8 | 7.3 |

因此,InterTris 的性能之所以表现较优,主要包括两个原因:第一,其为头实体、关系和尾实体分别构建了语义和映射向量,分开建模语义信息和交互过程;第二,其认为头实体、关系和尾实体三者中的任何一个都会同时受到另外两个的影响,在三元组粒度上实现了三者交互的充分建模.

表 7 是基于电子商务领域用户消费行为知识图谱样本 UserAct 的链接预测结果.首先,与 LiveIn 相比,UserAct 的整体 *Mean Rank* 较低,*Hit@k* 也相对较高,这是因为 UserAct 的数据集更为稠密,几乎所有模型都得到了更好的训练.其次,从 UserAct 数据集的内部看,InterTris 在头尾实体的 *filt* 预测,尤其是在 *Hit@10* 指标上,均达到了最优.紧随其后是 Complex 模型,其在尾实体的 *filt Mean Rank* 和 *raw Hit@10* 上达到了最优.最后,在进行头尾实体预测时,组合模型的效果要优于转换模型.但在关系预测中,*TransD* 却是表现最好的.这是因为其在建模过程中,充分考虑了实体-关系交互,具体表现为映射后的头尾实体均与关系相关.

综合上述实验结果,我们可以发现,就链接预测任务而言,由于在三元组粒度上考虑了知识图谱表示学习问题;同时,通过对头实体、关系和尾实体三者之间的交互进行了充分建模以保证特征抽取完备性,InterTris 模型可以在特定领域数据集上表现出最优的整体性能.

表 7　基于 UserAct 的链接预测结果

| 模型 | 头实体预测 | | | | 尾实体预测 | | | | 关系预测 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *Mean Rank* | | *Hit@10*（%） | | *Mean Rank* | | *Hit@10*（%） | | *Mean Rank* | | *Hit@10*（%） | |
| | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* | *raw* | *filt* |
| TransE[19] | 2106 | 1693 | **0.4** | 0.4 | 2033 | 2025 | 0.4 | 0.4 | 4 | 3 | 19.7 | 25.2 |
| TransH[21] | 2134 | 1699 | 0.3 | 0.3 | 1908 | 1899 | 0.1 | 0.1 | 3 | 3 | 12.9 | 20.9 |
| TransR[22] | 2135 | 1704 | 0.1 | 0.1 | 1440 | 1431 | 0.5 | 0.6 | 3 | 3 | 15.2 | 22.4 |
| TransD[23] | 1681 | 1241 | 0.3 | 0.4 | 133 | 124 | 10.5 | 12.8 | **2** | **1** | **43.3** | **96.4** |
| TranSparse[24] | 2109 | 1681 | **0.4** | 0.4 | 2037 | 2028 | — | — | 3 | 2 | 21.3 | 30.8 |
| DistMult[28] | 989 | 172 | — | 47.1 | 46 | 30 | 13.7 | 86.7 | 3 | **1** | 4.3 | 94.5 |
| HolE[29] | **925** | 396 | 0.1 | 1.0 | **41** | 30 | 3.9 | 14.6 | 3 | 3 | 29.4 | 29.4 |
| Complex[30] | 946 | 174 | — | 48.8 | 43 | **28** | **19.5** | 86.6 | **2** | **1** | 11.2 | 94.5 |
| ANALOGY[31] | 988 | 171 | — | 50.2 | 48 | 32 | 13.1 | 85.5 | 3 | **1** | 4.3 | 93.8 |
| InterTris | 946 | **158** | — | **52.9** | 54 | 39 | 18.6 | **87.6** | **2** | **1** | 10.0 | 93.8 |

### 4.3　三元组分类

三元组分类[27]用于判断三元组是否成立. 与已有工作[19-24]一样，本文评测指标为准确率(accuracy).

与链接预测任务类似，基于 Kinship 和 ES，InterTris的嵌入式表示维度为 20，其它训练参数为：学习率 0.1，最大训练轮数 1000. 调参范围为学习率衰减系数 $\gamma\in\{0.1,0.01,0.001\}$，负例个数 $n\in\{3,4,5,6\}$. 通过网格搜索，本文得到 Kinship 的最优参数为 $\gamma=0.01$，$n=5$；ES 的最优参数为 $\gamma=0.001$，$n=3$.

基于 LiveIn 和 UserAct，InterTris 训练参数为学习率 0.1 和最大训练轮数 1000. 调参范围为向量维度 $k\in\{20,100,150,200\}$，学习率衰减系数 $\gamma\in\{0.1,0.01,0.001,0.0001\}$，负例个数 $n\in\{3,4,5,6\}$. 通过网格搜索，LiveIn 的最优参数为 $k=200$，$\gamma=0.0001$，$n=5$；UserAct 的最优参数为 $k=50$，$\gamma=0.1$，$n=6$.

图 6 是三元组分类的实验结果. 无论是家谱领域的公共知识图谱 Kinship，还是微生物领域的酶知识图谱样本 ES，InterTris 模型都达到了最高的准确率，尤其是 ES 数据集，InterTris 的准确率比第二名 TranSparse 高了 17.69%. 但是，就 InterTris 模型而言，其在 Kinship 上取得了比 ES 更好的实验效果. 这是因为在 ES 数据集中，每个实体在训练集中平均出现了 155 417/57 066≈3 次；在 Kinship 数据集中，虽然训练数据集规模比较小，但是实体的平均出现次数可以达到 6411/104≈61 次. 也就是说，ES 的数据规模更大，但 Kinship 却更稠密. 由于考虑到头实体、关系和尾实体三元素之间的充分交互，InterTris 在 Kinship 数据集上的效果更好.

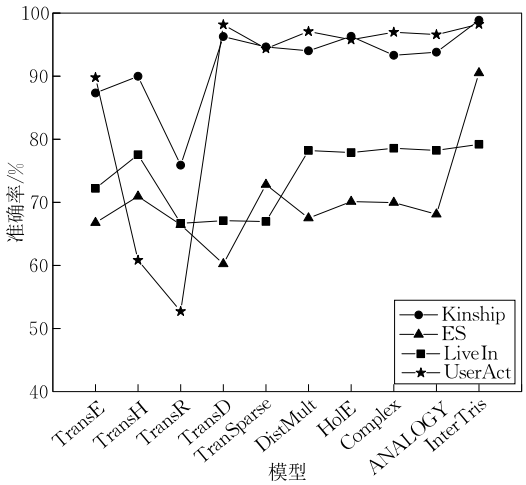


图 6　三元组分类实验结果

对于本文构造的数据集 LiveIn 和 UserAct 而言，InterTris 也都达到了最好的效果，分别为 79.21% 和 98.24%. 与 UserAct 相比，LiveIn 的三元组分类结果普遍较低. 这是因为 LiveIn 的数据分布较稀疏，几乎所有模型都未得到充分训练. 相比之下，组合模型的三元组分类效果与 InterTris 接近，而转换模型的效果则没有那么好. 这是因为组合模型对头实体、关系和尾实体三者之间的交互作用进行了较为充分的建模. 但是，TransD 和 TranSparse 在 UserAct 上的准确率比一般转换模型更好. 这是因为两者分别考虑了实体-关系之间的交互，以及关系异质性和不平衡性.

根据上述三元组分类实验结果，我们可以发现，就三元组分类任务而言，虽然并非针对某一具体的数据特征构建而成，InterTris 模型依旧具有更好的表达效果，且适应性更强. 这与其“三元组”级别的建模粒度，以及对头实体、关系和尾实体之间交互的充分拟合密切相关.

## 5　总　结

本文围绕特定领域的知识图谱表示学习问题，从机器学习视角出发对知识图谱表示学习问题进行

了分析，提出模型构建过程中的两个关键点，即目标数据集范围定义和特征抽取完备性. 然后，根据知识图谱构建语义联系的本质特征，考虑在语义基本单位三元组粒度上进行建模. 为保证特征抽取完备性，本文认为头实体、关系和尾实体三者中的任意一个都会同时受到另外两个因素的影响. 因此构建了三元交互的知识图谱表示学习模型 InterTris. 基于家谱领域的公共知识图谱 Kinship、微生物领域的酶知识图谱样本 ES、微生物领域的栖息地知识图谱样本 LiveIn 和电子商务领域的用户消费行为知识图谱样本 UserAct 共 4 个实验数据集，以部分转换模型和组合模型为基线，本文进行了链接预测和三元组分类两个实验任务. 结果表明，InterTris 可以在上述四个数据集上均表现出较优性能，说明了在三元组粒度，对头实体、关系和尾实体三者之间交互过程进行充分建模的必要性和合理性.

为进一步改善知识图谱表示学习效果，本文提出如下三点研究方向：

(1) 负例生成方式的改进. 最简单的负例生成方式是基于 Closed World 假设[36] 的 unif[19]. 而 Wang 等人[21] 为了降低生成假负例的概率，提出 bern 方法. 该方法认为，如果相同头实体和关系对应的尾实体越多，那么在为该三元组生成负例时，就应当考虑更大概率去替换头实体，这样就可以减少假负例的生成. 但是，这样的负例却很难将正负例尾实体区分开，导致模型的表达能力受限. 因此，假负例和模型表达能力是两个相互制约的因素，如何在两者之间达到平衡是一个有待考虑的问题. 目前已经出现了基于 GAN 框架的负例生成方式[37]，可以进一步提升训练效果.

(2) 本体层信息的充分利用. 嵌入式知识图谱表示学习往往仅对数据层信息进行学习. 但知识图谱还包括本体层. 早在知识图谱出现之前的语义网时代，“本体层”概念就已诞生. 所以，目前有很多本体层链接[38] 和推理[39-41] 等方面的工作. 如果可以进一步将本体层中的信息加入到嵌入式表示过程中，进而体现在数据层中每个实体和关系的数值表示中，那么表示学习结果的可用性就越强. 同时，也可以考虑将本体层知识推理的结果应用到相应数据层中. 由于本体层规模一般都小于数据层，且本体层中的一个节点或关系往往对应更高数量级的数据层节点或关系，所以这种推理方式的计算成本较低，但收益却很高.

(3) 外部知识的充分利用. 虽然知识图谱表示学习可以用于推理，进而用于知识图谱的计算补全过程. 但是，计算补全本身存在无法引入新实体和新关系等缺陷. 而且并非所有信息都可通过计算补全完成，尤其是身高、体重和出生地等人口属性. 可这些关系的缺失又非常严重. Freebase 中 71% 的人都没有出生地信息，而且这些人大多为明星和政治家等名人. 因此，需要考虑使用文本或其它数据源进行填充补全，从而奠定计算补全的基础.

## 参 考 文 献


[1] Suchanek F M, Weikum G. Knowledge harvesting in the big-data era//Proceedings of the 40th ACM Special Interest Group on Management of Data. New York, USA, 2013: 933-938

[2] Suchanek F M, Weikum G. Knowledge bases in the age of big data analytics. Proceedings of the Very Large Data Bases Endowment, 2014, 7(13): 1713-1714

[3] Meng Xiao-Feng, Du Zhi-Juan. Research on the big data fusion: Issues and challenges. Journal of Computer Research and Development, 2016, 53(2): 231-246(in Chinese)
(孟小峰，杜治娟. 大数据融合研究：问题与挑战. 计算机研究与发展，2016，53(2)：231-246)

[4] Zhang Y, He S, Liu K, et al. A joint model for question answering over multiple knowledge bases//Proceedings of the 30th Association for the Advancement of Artificial Intelligence. Phoenix, USA, 2016: 3094-3100

[5] Yang S, Zou L, Wang Z, et al. Efficiently answering technical questions—A knowledge graph approach//Proceedings of the 31st Association for the Advancement of Artificial Intelligence. San Francisco, USA, 2017: 3111-3118

[6] Zhang Y, Dai H, Kozareva Z, et al. Variational reasoning for question answering with knowledge graph//Proceedings of the 32nd Association for the Advancement of Artificial Intelligence. New Orleans, USA, 2018: 6069-6076

[7] Dong X, Gabrilovich E, Heitz G, et al. Knowledge Vault: A web-scale approach to probabilistic knowledge fusion//Proceedings of the 20th ACM Special Interest Group on Knowledge Discovery and Data Mining. New York, USA, 2014: 601-610

[8] Zhang F, Yuan N J, Lian D, et al. Collaborative knowledge base embedding for recommender systems//Proceedings of the 22nd ACM Special Interest Group on Knowledge Discovery and Data Mining. San Francisco, USA, 2016: 353-362

[9] Shi C, Liu S, Ren S, et al. Knowledge-based semantic embedding for machine translation//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin, Germany, 2016: 2245-2254

[10] Bollacker K, Evans C, Paritosh P, et al. Freebase: A collaboratively created graph database for structuring human

knowledge//Proceedings of the 35th ACM Special Interest Group on Management of Data. Vancouver, Canada, 2008: 1247-1250

[11] Suchanek F M, Kasneci G, Weikum G. YAGO: A large ontology from Wikipedia and WordNet. Web Semantics, 2008, 6(3): 203-217

[12] Vrandečić D, Krötzsch M. WikiData: A free collaborative knowledge base. Communications of the ACM, 2014, 57(10): 75-85

[13] Auer S, Bizer C, Kobilarov G, et al. DBPedia: A nucleus for a web of open data//Proceedings of the 6th International Semantic Web Conference. Busan, R. O. Korea, 2007: 722-735

[14] Carlson A, Betteridge J, Kisiel B, et al. Toward an architecture for never-ending language learning//Proceedings of the 24th Association for the Advancement of Artificial Intelligence. Atlanta, USA, 2010: 1306-1313

[15] Wu W, Li H, Wang H, et al. Probase: A probabilistic taxonomy for text understanding//Proceedings of the 39th ACM Special Interest Group on Management of Data. Scottsdale, USA, 2012: 481-492

[16] Belleau F, Nolin M A, Tourigny N, et al. Bio2RDF: Towards a mashup to build bioinformatics knowledge systems. Journal of Biomedical Informatics, 2008, 41(5): 706-716

[17] Ashburner M, Ball C A, Blake J A, et al. Gene ontology: Tool for the unification of biology. Nature Genetics, 2000, 25: 25

[18] Kemp C, Tenenbaum J B, Griffiths T L, et al. Learning systems of concepts with an infinite relational model//Proceedings of the 21st National Conference on Artificial Intelligence. Boston, USA, 2006: 381-388

[19] Bordes A, Usunier N, Garcia-Duran A, et al. Translating embeddings for modeling multi-relational data//Proceedings of the Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems. Lake Tahoe, USA, 2013: 2787-2795

[20] Bordes A, Weston J, Collobert R, et al. Learning structured embeddings of knowledge bases//Proceedings of the 25th Association for the Advancement of Artificial Intelligence. San Francisco, USA, 2011: 301-306

[21] Wang Z, Zhang J, Feng J, et al. Knowledge graph embedding by translating on hyperplanes//Proceedings of the 28th Association for the Advancement of Artificial Intelligence. Quebec, Canada, 2014: 1112-1119

[22] Lin Y, Liu Z, Sun M, et al. Learning entity and relation embeddings for knowledge graph completion//Proceedings of the 29th Association for the Advancement of Artificial Intelligence. Austin, USA, 2015: 2181-2187

[23] Ji G, He S, Xu L, et al. Knowledge graph embedding via dynamic mapping matrix//Proceedings of the 53rd Association for Computational Linguistics. Beijing, China, 2015, 1: 687-696

[24] Ji G, Liu K, He S, et al. Knowledge graph completion with adaptive sparse transfer matrix//Proceedings of the 30th Association for the Advancement of Artificial Intelligence. Phoenix, USA, 2016: 985-991

[25] Nickel M, Tresp V, Kriegel H P. A three-way model for collective learning on multi-relational data//Proceedings of the 28th International Conference on Machine Learning. Bellevue, USA, 2011: 809-816

[26] Jenatton R, Roux N L, Bordes A, et al. A latent factor model for highly multi-relational data//Proceedings of the Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems. Lake Tahoe, USA, 2012: 3167-3175

[27] Socher R, Chen D, Manning C D, et al. Reasoning with neural tensor networks for knowledge base completion//Proceedings of the 26th Advances in Neural Information Processing Systems: the 27th Annual Conference on Neural Information Processing Systems. Lake Tahoe, USA, 2013: 926-934

[28] Yang B, Yih S W-T, He X, et al. Embedding entities and relations for learning and inference in knowledge bases//Proceedings of the 3rd International Conference on Learning Representations. San Diego, USA, 2015

[29] Nickel M, Rosasco L, Poggio T. Holographic embeddings of knowledge graphs//Proceedings of the 30th Association for the Advancement of Artificial Intelligence. Phoenix, USA, 2016: 1955-1961

[30] Trouillon T, Welbl J, Riedel S, et al. Complex embeddings for simple link prediction//Proceedings of the 33rd International Conference on Machine Learning. New York, USA, 2016: 2071-2080

[31] Liu H, Wu Y, Yang Y. Analogical inference for multi-relational embeddings//Proceedings of the 34th International Conference on Machine Learning. Sydney, Australia, 2017: 2168-2178

[32] Bordes A, Glorot X, Weston J, et al. A semantic matching energy function for learning with multi-relational data. Machine Learning, 2014, 94(2): 233-259

[33] Duchi J, Hazan E, Singer Y. Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research, 2011, 12(7): 257-269

[34] Niu F, Recht B, Re C, et al. Hogwild!: A lock-free approach to parallelizing stochastic gradient descent//Proceedings of the Advances in Neural Information Processing Systems 24: the 25th Annual Conference on Neural Information Processing Systems. Granada, Spain, 2011: 693-701

[35] Zhang Y, Du Z, Meng X. EMT: A tail-oriented method for specific domain knowledge graph completion//Proceedings of the 23rd Pacific Asia Knowledge Discovery and Data Mining. Macau, China, 2019: 514-527

[36] Reiter R. On closed world data bases. Readings in Artificial Intelligence, 1981: 119-140

[37] Wang P, Li S, Pan R. Incorporating GAN for negative s ampling in knowledge representation learning//Proceedings of the 32nd Association for the Advancement of Artificial Intelligence. New Orleans, USA, 2018: 2005-2012

[38] Mohammadi M, Hofman W, Tan Y. A comparative study of ontology matching systems via inferential statistics. IEEE Transactions on Knowledge and Data Engineering, 2019, 31(4): 615-628

[39] Galárraga L, Teflioudi C, Hose K, et al. Fast rule mining in ontological knowledge bases with AMIE+. The Very Large Data Base Journal, 2015, 24(6): 707-730

[40] Liang J, Xiao Y, Wang H, et al. Probase+: Inferring missing links in conceptual taxonomies. IEEE Transactions on Knowledge Data Engineering, 2017, 29(6): 1281-1295

[41] Fernández-Álvarez D, García-González H, Frey J, et al. Inference of latent shape expressions associated to DBPedia ontology//Proceedings of the 17th International Semantic Web Conference 2018 Posters & Demonstrations. Monterey, USA, 2018

**ZHANG Yi**, M. S. candidate. Her main interests include knowledge graph building and representation learning.

**MENG Xiao-Feng**, Ph. D. supervisor, professor. His main interests include big data fusion (knowledge graph building), big data real-time analysis, big data privacy protection and interdisciplinary researches like social computing.



**Background**

The problem discussed in this paper is knowledge graph representation learning, which belongs to knowledge graph domain. As an emerging data collection and organization technology, knowledge graph models the real world in symbolized form. To achieve further value mining of knowledge graph, we need to represent the symbolized form as numerical one. Therefore, knowledge graph representation learning arose. Up to now, knowledge graph representation learning has been developed for almost 10 years. The related models include translation one, composition one and neural network one. Based on the application domain, knowledge graph can be divided into open domain and specific domain. Although developed well, nearly all the existing models are inspired by data distributions in the open domain knowledge graph, and constructed for open domain, finally verified in open domain experimental dataset. When applied on specific domain knowledge graph, they will be challenged by new data distributions. So, this paper aims to alleviate the challenges in specific domain knowledge graph representation learning.

Based on specific domain dataset analysis, we found that data distribution varies even among specific domain knowledge graphs. Therefore, from the perspective of more abstract than data distributions, according to the semantic connection construction essence of knowledge graph, taking a triplet as the granularity, this paper put forward InterTris by modeling the interaction among head entity, relation and tail entity. On the one hand, the triplet granularity ensures that our model depends on the existence of a triplet but a sub-graph. All of these knowledge graphs have the same triplet structure, i.e. head entity, relation, tail entity, but different sub-graphs. On the other hand, the interaction modeling can cover three cases: the same head and relation reach different tails, the same head and tail are connected by different relations, the same relation and tail is shared by different heads. So, theoretically, InterTris can model specific knowledge graph well.

Also, experimentally, taking some better translation and composition models as baseline, based on the public knowledge graph Kinship in genealogy, the enzyme knowledge graph sample ES in microbiology, the habitat knowledge graph sample LiveIn in microbiology, and the consumer behaviors knowledge graph sample UserAct in e-commerce, this paper carried out two experimental tasks, i.e. link prediction and triplets classification, showing that InterTris has the best overall effect, proving the necessity and rationality of the triplet granularity and the interaction modeling.

This work was partially supported by the grants from the National Natural Science Foundation of China (Nos. 91846204, 61941121).