文章编号：1003-0077(2020)05-0091-09

双特征空间的实体排序学习

赵以昕[1]，牛树梓[1]，纪春岩[2]，卢 菲[2]，徐 睿[3]

(1. 中国科学院 软件研究所，北京 100190；
2. 山东大学 齐鲁医院，山东 济南 250012；
3. 中科嘉速(北京)信息技术有限公司，北京 100190)

摘 要：随着大规模知识图谱的出现以及企业高效管理领域知识图谱的需求，知识图谱中的自组织实体检索成为研究热点。给定知识图谱以及用户查询，实体检索的目标在于从给定的知识图谱中返回实体的排序列表。从匹配的角度来看，传统的实体检索模型大都将用户查询和实体统一映射到词的特征空间。这样做具有明显的缺点，例如，将同属于一个实体的两个词视为独立的。为此，该文提出将用户查询和实体同时映射到实体与词两个特征空间方法，称为双特征空间的排序学习。首先将实体抽象成若干个域。之后从词空间和实体空间两个维度分别抽取排序特征，最终应用于排序学习算法中。实验结果表明，在标准数据集上，双特征空间的实体排序学习模型性能显著优于当前先进的实体检索模型。

关键词：知识图谱；实体检索；双特征空间

中图分类号：TP391 **文献标识码**：A

Learning to Rank Entities from Dual Feature Spaces

ZHAO Yixin[1], NIU Shuzi[1], JI Chunyan[2], LU Fei[2], XU Rui[3]

(1. Institute of Software, Chinese Academy of Sciences, Beijing 100190, China;
2. Qilu Hospital of Shandong University, Jinan, Shandong 250012, China;
3. Sinoparasoft Company Limited, Beijing 100190, China)

Abstract: Entity retrieval from knowledge graph is of substantial significance as the large scale knowledge graphs appear, and the industry demand on effectively managing the domain knowledge graphs. Given a certain knowledge graph and a user query, entity retrieval aims at obtaining a ranking list of entities from the knowledge graph according to its relevance to the query. Being treated as the matching between the query and entities, traditional entity retrieval models map both user queries and entities into the word feature space. However, it does not work when two words in the name of an entity are assumed to be independent. In this paper, we propose to project both user queries and entities into a dual feature space, namely entity-word feature space. First, we represent entities as multiple domains and extract ranking features from them. Then, learning to rank models are employed to train a ranking model from this dual feature space. Experimental results on benchmark datasets show that our proposed method outperform state-of-the-art baselines significantly.

Keywords: knowledge graph; entity retrieval; dual features space

0 引言

为了从海量信息中挖掘到的知识进行结构化的组织，知识图谱应运而生。近年来，以 DBpedia[①]，Freebase[②] 为代表的大型开放知识库的出现推动了知

① https://wiki.dbpedia.org/
② https://developers.google.com/freebase

收稿日期：2019-10-29 **定稿日期**：2019-11-09

基金项目：国家自然科学基金(61602451)；山东省重大科技创新工程项目(2018CXGC1215)；山东大学基本科研业务费资助项目(26010177611018)

识图谱相关技术的发展，知识库中承载的丰富语义信息可以用于深层次的文档理解、表示有关的任务。知识图谱可以从一定程度上解决信息过载问题，提供更为直观的使用方式，为知识驱动的任务提供支持[1]。因此，知识图谱的构建及其应用已经成为研究热点。给定自然语言的用户查询输入，如何从知识图谱中找到相关的实体是基于知识图谱的实体检索研究的核心问题。

如何实现自然语言文本(无结构)与结构化文本之间的匹配是基于知识图谱的实体检索面临的主要挑战。直观的做法是把实体的结构压缩为一段文本，从而将问题简化为纯文本的匹配。更通用的做法是将实体结构抽象为若干域，并将查询和每个域进行文本匹配得到最终的排序得分。本文将此类检索方法称为域模型(field model)。典型的域模型，如FSDM[2]将实体抽象为实体名称、属性、所属类别、相似实体名集合、相关实体名集合等五个域。其他域模型包括BM25F[3]、MLM[4]等。域模型的优点是没有将实体做完全扁平化处理，在实际应用中与知识图谱嵌入[5]相比能取得更为持续稳定的效果，这在对精度要求较高的领域，如医学领域，是很好的选择。域模型的缺点是并没有充分利用知识图谱。

自然语言文本中最基本的单元是词，但是知识图谱中最基本的单元是实体。因此，实体检索所面临的另一个挑战就是自然语言文本无法直接与实体匹配。传统的实体检索模型，如FSDM与BM25F等的做法是都统一将词作为基本单元。这样就造成了对实体表示的不足。例如，用户查询“红细胞沉降率”(erythrocyte sedimentation rate)在词空间被划分为“erythrocyte、sedimentation、rate”。根据这三个词直接定位到实体“erythrocyte sedimentation rate”并不容易，因为同时包含这三个词的实体描述太多。

将用户查询与实体都统一表示到实体空间如何？用户查询为自然语言的文本，有时可能并不包含实体。若把用户查询直接映射到实体空间，会导致查询表示过于稀疏。因此，本文提出了实体—词双特征空间的实体排序学习框架。

首先，将用户查询的自然语言文本进行实体链接，获得查询在实体空间的表示。其次，将实体在知识图谱中的结构抽象成 t 个域。再次，获得查询与实体各个域之间的实体—词双特征空间表示。最后，利用排序学习算法学习各个域在每个特征空间的权重，综合得分作为最终的排序依据，返回相关实体排序列表。

为了验证模型的有效性，本文在DBpedia以及医学数据集OHSUMED上进行实验。结果表明，双特征空间实体排序学习框架明显优于当前最先进的域模型。

1 相关工作

依据双特征空间的实体排序学习框架的实现步骤，本文从检索模型以及实体链接两个角度来简述相关工作。

1.1 检索模型

已有的纯文本检索模型、域模型以及基于知识图谱嵌入的检索模型都可以用于实体检索任务。

1.1.1 纯文本检索模型

空间向量模型(VSM)[6]是文本检索早期代表性的理论之一，主要思想是将文档建模到词项空间中，并基于空间距离描述文档和查询的相似度。在此基础上，进一步提出了区分词项重要性的TFIDF加权技术。然而基于词项的文本表示方法无法描述文本的语义信息，最终会面临相关文档和查询之间词汇不匹配的问题。为了克服这种缺陷，潜在语义索引[7](latent semantic indexing)通过奇异值分解对词频向量降维，将文档映射到稠密的潜在语义空间，从而建模文档之间的语义关系。此外，另一种研究视角使用了概率模型建模，将查询和文档是否相关的概率作为检索系统的排名依据。BM25[8]和基于统计语言模型的似然检索模型LM[9]是其中最有代表性的理论。SDM[10]将词序(顺序依赖)引入到语言模型中以期达到更好的检索效果。

除上述提到的传统方法之外，还有一类从用户相关反馈中学习规律的方法，被称为排序学习。传统的排序学习基于人工设计的特征构建训练数据，根据不同的学习策略分为基于文档和查询相关性的pointwise、基于文档偏序关系的pairwise、基于查询与文档列表关系的listwise三种方式。其中代表性的模型有基于pairwise的RankSVM[11]、RankNet[12]，基于listwise的ListNet[13]、LambdaMRT[14]等。

此外，近年来出现了许多基于深度神经网络的排序学习模型。这类方法能够自动从训练数据中学习有意义的潜在特征和结构，从而弥补人工提取特

征的各种缺陷。深度语义匹配模型[15](deep structured semantic models)将原始的查询和文档以n-gram的形式输入，通过全连接网络映射到同一个表示空间，并计算相似度作为排序结果。DRMM[16](deep relevance matching model)在此基础上，加入预训练的词嵌入表示进行改进，以直方图的形式将词级别的相似度通过门控网络进行聚合。

1.1.2　域模型

MLM[4]和FSDM[2]均使用了线性组合的方式利用结构化文档中的每个域。其中前者分析了元搜索算法融合的关键问题，在基于语言模型的查询似然检索模型[9]的基础上提出了MLM[4](mixture of language models)模型。后者在此基础上结合了序列依赖模型进行改进，提出了FSDM[2](fielded sequential dependence model)。另外，此项工作还提出了一种适用于实体检索的实体多域表示方案，从而更好地捕获实体和实体关系的语义信息。PFSDM[17](parameterized fielded sequential dependence model)进一步考虑了查询词的重要性，从而更准确地捕获用户的查询意图。

BM25F[3]将BM25[8]扩展到了结构化文档，分析了传统做法对于不同域加权结合的潜在缺陷。BM25F[3]不再为每一个域单独计算一个排名分数再进行结合，而是先为BM25[8]中的词频、文档长度等基本统计量进行加权合并，再带入到原始排名函数中。appLDA[18]提出了一种联合建模不同文本域的概率主题模型。该模型捕获的主题可以同时反映出不同域各自的词项分布特点。

1.1.3　基于知识图谱嵌入的检索模型

ESR(explicit semantic ranking)模型[19]提出了一种利用知识图谱中的语义信息将查询和文档相关联的方法。ESR模型[19]使用知识图谱的实体嵌入捕捉查询和文档的匹配信号，并通过软匹配与完全匹配两种方式构建排序特征，最终通过排序学习模型进行组合。在此基础上，文献[20]提出的AttR-Duet将匹配信号进一步扩展为词空间与实体空间的交互排序特征，并加入了注意力机制进行改进。MEMBER[21]是一种注重可解释性的实体嵌入方法，能够将实体与词以不同的几何形式嵌入到同一表示空间中，并基于词与实体的空间关系构造了直观的排名函数。

1.2　实体链接

实体链接是一种从文本中抽取实体的技术，可以帮助在输入文本中标记出简单而有意义的术语序列。目前已经有多种成熟而有效的实体链接系统解决了此类问题。文献[22]提出了一种为短文本设计的实体链接系统，通过n-gram生成候选实体的列表，之后基于监督学习的方法进行消岐。GLOW系统[23]考虑了实体消岐中实体指称全局一致性的问题，使用了Illinois NER识别文本中的实体指称，为实体指称构建局部和全局的排序特征，并通过排名函数计算指称和页面的相关性。TagMe系统[24]首先为维基百科中的锚点，重定向实体指称和实体名构建索引，通过精确匹配识别文本中的指称，之后设计了一种基于局部似然的投票方案，最大化文本中实体之间的一致性。

此外，文献[25]中提出了一种评估实体链接系统的基准框架，并对几种主流系统进行了比较。最终TagMe[24]在实体链接、实体识别两方面均达到了最佳性能，被认为是最理想的实体注释系统。因此本文选择了TagMe[24]用于抽取文本中的实体。

2　双特征空间的实体排序学习

首先对基于知识图谱的实体检索问题及相关模型形式化，接下来介绍本文提出的双特征空间的实体排序学习框架，最后介绍具体的算法实现。

2.1　形式化

给定知识图谱$G=(E,R,T)$，其中E为实体集合，R为关系集合，$T=\{(e_h, r, e_t)\}\subset E\times R\times E$为图谱中的三元组集合，$e_h\in E$，$e_t\in E$，$r\in R$。任一用户查询$q$是自然语言的词串。实体检索的目标在于学习到查询与实体之间的匹配函数$f(q, e)$。

位于图谱中的任一实体e，可用与之有关系的实体或属性等构成的子图来表示。为了简化，本文对e的表示做不完全的扁平化处理，即将e相关的子图分解为t个域$F=\{F_i|i=1,\cdots,t\}$，每个域F_i表示与e具有一类关系f_i的实体名称集合$F_i=\{e'.\text{name}|(e, r, e')\text{ or }(e', r, e)\text{ and }r\in f_i\}$。例如，FSDM中的$f_i$是名称、属性、类别、相似关系、相关关系等。至此，域模型将$f(q, e)$转换为$f(q, \{F_i|i=1,\cdots,t\})$。传统的域模型，如BM25F[3]、FSDM[2]对匹配函数的定义都满足式(1)：

$$f(q,e)=\sum_{i=1}^{t}\lambda_i \boldsymbol{w}_i\cdot\boldsymbol{\Phi}_w(q,F_i) \tag{1}$$

其中，$\boldsymbol{\Phi}_w(q, F_i)\in R^d$表示查询$q$与实体$e$的域

F_i同时映射到词空间的特征表示，λ_i表示该域的权重。

2.2 框架

为了解决查询与实体之间的匹配带来的挑战，同时具备稳定可靠的检索效果，本文引入双特征空间映射 $\boldsymbol{\Phi}_{we}(q, F_i)$，提出了双特征空间的实体排序学习(learning to rank entities from dual feature space)，简记为 D-L2RE。

任一查询 q 的待排序实体集合 $X_q=\{x_1, x_2, \cdots, x_n\}$，其中，$x_i$表示实体 e_i的在双特征空间的向量表示。$Y_q=\{y_1, y_2, \cdots, y_n\}$表示相关性标注的集合，$y_i$表示查询 q 与实体 e_i的相关程度标签，如采用三级相关性标注 $y_i \in \{0$(不相关)，1(部分相关)，2(相关)$\}$。给定训练集$\{(X_q, Y_q)\}$，优化如公式(2)所示的目标函数，学习排序函数 $f_{we}(q, e)$，其中$L(Y_q, f_{we}(\omega, X_q))$表示损失函数。

$$\min \sum_q L(Y_q, f_{we}(\omega, X_q)) \tag{2}$$

为了简化，双特征空间映射定义为词空间与实体空间的连接操作，如式(3)所示。后续工作中会引入 max-pooling、线性组合等其他操作来进一步验证双特征空间的有效性。

$$\boldsymbol{\Phi}_{we}(q, F_i)=[\boldsymbol{\Phi}_w(q, F_i), \boldsymbol{\Phi}_e(q, F_i)] \tag{3}$$

本文采用不包含偏置的简单的线性神经网络作为排序函数，定义如式(4)所示。用 $\boldsymbol{\Phi}_{we}(q, F_i)$代替式(1)中 $\boldsymbol{\Phi}_w(q, F_i)$，将权重合并，最终可以得到如式(3)所示的线性模型。不同的是，本文提出的框架中的权重通过排序学习算法训练得到。从这个意义上讲，本文提出的双特征空间实体排序学习框架是对域模型进行了扩展。

$$f_{we}(\boldsymbol{\omega}, X_q)=\boldsymbol{\omega} \cdot \boldsymbol{\Phi}_{we}(q, F) \tag{4}$$

2.3 算法

本文将双特征空间的实体排序学习框架的实现划分为如图 1 所示的四个步骤：①实体不完全扁平化处理：从知识图谱中抽出与实体 e 相关的子图，并按关系类别将结构归并为 t 个域；②查询实体链接：对用户查询文本进行命名实体识别与消歧；③双特征空间映射：从词空间与实体空间两个维度分别抽取实体各个域的特征；④排序学习算法：这里以 pairwise 学习算法 RankSVM[11] 为例。

2.3.1 实体链接

查询 $q=w_1 w_2 \cdots w_m$以自然语言文本的形式存在，很自然地可以得到词空间的表示，以 bag-of-words 模型[6]为例，$B_w(q)=\{(w, c(w, q))\}$，$c(a, b)$表示 a 在 b 中出现的次数。为了获得查询在实体空间的表示，本文采用命名实体识别与链接技术来实现。首先，对查询文本进行实体识别，找到输入文本中所有可能表示某个实体概念的实体指称(entity mention)。其次，由于不同的实体概念可能具有相同的实体指称，所以需要对实体指称进行消歧处理。

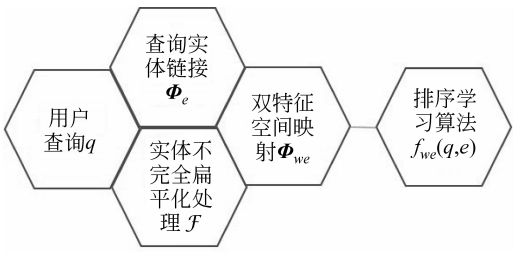

图 1 双特征空间实体排序学习

本文使用目前主流的实体链接工具 TagMe[24] 来抽取查询文本中的实体。TagMe[24] 利用维基百科页面中锚点的超链接信息，通过精确匹配的方式找到实体指称集合，综合考虑实体的先验链接概率以及文本中的全局因素，共同计算每个候选实体的最终链接概率。通过实体链接，得到查询的实体表示，以 bag-of-entities[26] 模型为例，$B_e(q)=\{(e, c(e, q))\}$。

2.3.2 双特征空间映射

双特征空间映射 $\boldsymbol{\Phi}_{w,e}(q, F_i)$的定义如式(3)所示，其中 $\boldsymbol{\Phi}_w(q, F_i)$表示将查询与实体的域 F_i在词空间匹配的特征向量，$\boldsymbol{\Phi}_e(q, F_i)$表示将查询与实体的域 F_i在实体空间匹配的特征向量。

具体来讲，若将 q 与 F_i 都表示为 bag-of-words[6] 模型，$\boldsymbol{\Phi}_w(q, F_i)$可以定义为 $1\times d_w$ 维向量。参考 LETOR[27] 数据集的特征向量定义，每一维的定义如表 1 所示，这里 $d_w=12$。需要说明的是，在语言模型相关特征中本文采用的都是一元(unigram)语言模型，涉及 $P(q \mid F_i)$的计算如式(5)所示。

$$\begin{aligned} P(q \mid F_i) &= \prod_{w \in B_w(q)} P(w \mid P_i), \\ P(w \mid F_i) &= \frac{c(w \mid F_i)}{\mid F_i \mid} \end{aligned} \tag{5}$$

类似地，若将 q 与 F_i 都表示为 bag-of-entities[26] 模型，$\boldsymbol{\Phi}_e(q, F_i)$可以定义为 $1\times d_e$ 维向量。不同之处在于，实体在查询文本中出现的次数往往

表 1　查询与实体域在词空间的特征向量

维度	名称
1	Occurrence_count
2	log(Occurrence_Count)
3	TF
4	log(TF)
5	IDF
6	log(IDF)
7	TF-IDF
8	Log(TF-IDF)
9	Language Model
10	Language Model with JM Smoothing
11	Language Model with ABS Smoothing
12	Language Model with Dirichlet Smoothing

过低，容易造成数据稀疏。因此，本文在构建实体级别的语言模型时，引入了实体链接 TagMe[24] 得到的置信度。这样，查询中实体出现在实体域 F_i 中的次数用查询中所有实体指称的链接置信度来估计，如式(6)所示。M_e 表示实体指称集合中注释中包含实体 e 的指称集合。进一步，实体域 F_i 的长度用其中出现的每个实体 e' 的频次估计函数之和来定义，如式(7)所示。

$$\hat{c}(e,F_i)=\boldsymbol{E}[c(e,F_i)]=\sum_{e'\in M_e}P_{e'} \tag{6}$$

$$|F_i|=\sum_{e'\in B_e(F_i)}\hat{c}(e',F_i) \tag{7}$$

对于实体的每个域 F_i，采用词空间的特征映射，得到查询与实体的域之间的 $1\times d_w$ 维的特征向量；采用实体空间的特征映射，得到查询与实体的域之间的 $1\times d_e$ 维的特征向量。两者连接，可以得到每个域在双特征空间的向量表示为 $1\times(d_w+d_e)$ 维。故查询与实体在双特征空间的向量表示为 $1\times(d_w+d_e)t$ 维。

2.3.3　排序学习算法 RankSVM

本文采用排序学习算法来学习实体检索模型中如式(4)所示的特征权重 ω。基于 pairwise 的排序学习算法是其中抗噪声能力好、性能稳定的一类。本文选取典型代表 RankSVM[11] 来验证双特征空间的实体排序学习框架的有效性。不同 pairwise 算法的主要差异在于式(2)中损失函数定义不同。本文采用的 RankSVM[11] 的损失函数 $L(Y_q, f_{we}(\omega, X_q))$ 如式(8)所示。本文采用 cutting-plane 算法来优化目标函数，具体实现在工具包①中。

$$L=\frac{1}{2}\|\boldsymbol{\omega}\|^2+C\sum_{i=1}^{n}\sum_{j=1,j\neq i}^{n}\max(0,1-\boldsymbol{\omega}\cdot(x_i-x_j))I(y_i>y_j) \tag{8}$$

3　实验结果

本文从性能、特征选择与收敛速度等方面来分析双特征空间实体排序学习算法 D-L2RE 的有效性。

3.1　实验设置

数据集　实验采用了 DBpedia[28] 与 OHSUMED[27] 两个标准的实体检索数据集。DBpedia 是从维基百科中抽取出来的结构化的信息，是迄今为止网络上存在的最大的综合型知识库。DBpedia 3.7 包含了 364 万的实体。面向实体的查询需求集合是多个竞赛任务的合集。经过一些数据清洗的步骤，最终确定了 485 个查询，每个查询约有 27 个相关实体。OHSUMED 数据集的实体集是医学刊物数据库 MEDLINE 的一个子集，包含约 30 万个来自 270 个不同医学期刊的记录。每个记录由标题、摘要、MeSH 索引项、作者、来源及刊物类型等组成。查询集合中的 106 个查询，涵盖了不同的医学检索需求，包括病人信息查询、医学主题查询等。实体与查询的相关程度由人工标注得到，采用三级标注：0(不相关)，1(部分相关)，2(相关)。数据集中包含 16 140 查询—实体对。

实体不完全扁平化处理　针对 DBpedia 数据集，本文从实体子图抽取出名称、属性、所属类别三个域。针对 OHSUMED 数据集，本文采用标题、摘要两个域。

基准方法　实验对比了文本检索模型与域检索模型两类方法。① 文本检索模型：BM25[8] 与 SDM[10]；② 域检索模型：BM25F[3] 与 FSDM[2]；③扩展的域检索模型：采用排序学习算法来学习域权重的模型，如仅采用词特征空间的实体排序学习

① https://www.cs.cornell.edu/people/tj/svm_light/svm_rank.html

算法 W-L2RE 与仅采用实体特征空间的实体排序学习算法 E-L2RE。本文提出的 D-L2RE 算法也属此类。

评价指标 NDCG[29]（normalized discounted cumulative gain）与 MAP[6]（mean average precision）用来评价检索模型的性能，如式(9)所示。

$$\begin{aligned} \mathrm{DCG}@k &= \sum_{i=1}^{k} \frac{2^{r_i}-1}{\log_2(1+i)} \\ \mathrm{NDCG}@k &= \frac{\mathrm{DCG}@k}{\mathrm{idealDCG}@k} \end{aligned} \tag{9}$$

其中，idealDCG@k 表示用相关性标注作为排序标准得到的 DCG@k，r_i 表示在按预测函数排序的列表中排在第 i 位的实体的相关性标注，如式(10)所示。

$$\mathrm{AP} = \frac{1}{n_r}\sum_{k=1}^{n} \frac{\sum_{i=1}^{k} I(r_i=1)}{k} I(r_k=1) \tag{10}$$

AP（average precision）针对 0（相关）-1（不相关）标注，定义了查询返回的排序列表的平均准确度。式(10)中 $I(r_i=1)$ 为指示函数，表示排在预测的排序列第 i 个位置的实体与查询是否相关，即第 i 个位置的实体的相关性标注是否为 1，相关返回 1，否则为 0。MAP 是对测试集中所有查询的 AP 的平均。

实验流程 实验中所呈现的性能都是在每个数据集上进行五折交叉验证的结果。其中，RankSVM 的超参数 C 从[0.001, 10]区间中选择最优值。

3.2 性能分析

根据上述实验流程，本文选取的纯文本检索、域检索以及扩展的域检索等三类模型中具有代表性的算法在 DBpedia 与 OHSUMED 上进行性能对比，分别如表 2、表 3 所示。

表 2 DBpedia 数据集上实体检索模型性能

模型	N@1	N@5	N@10	MAP
BM25	0.386 5	0.395 3	0.407 6	0.330 6
SDM	0.470 4	0.445 2	0.428 6	0.325 2
BM25F	0.503 2	0.458 6	0.457 2	0.353 6
FSDM	0.486 2	0.462 1	0.460 3	0.358 4
W-L2RE	0.494 2	0.464 8	0.457 6	0.351 2
E-L2RE	0.528 6	0.491 0	0.481 2	0.358 0
D-L2RE	**0.576 6**$^{\uparrow}$	**0.518 6**$^{\uparrow}$	**0.527 4**$^{\uparrow}$	**0.393 8**$^{\uparrow}$

注：↑采用 t-test 显著性检验，结果表明性能显著高于其他方法（$p<0.01$）。

表 3 OHSUMED 数据集上实体检索模型性能

模型	N@1	N@5	N@10	MAP
BM25	0.252 4	0.248 1	0.239 3	0.155 6
SDM	0.191 2	0.169 8	0.142 4	0.060 2
BM25F	0.314 2	0.296 1	0.293 2	0.177 4
FSDM	0.216 0	0.181 2	0.183 8	0.131 7
W-L2RE	0.267 1	0.237 8	0.228 5	0.138 1
E-L2RE	0.368 2	0.315 8	**0.312 6**$^{\uparrow}$	**0.183 6**$^{\uparrow}$
D-L2RE	**0.403 2**$^{\uparrow}$	**0.322 8**$^{\uparrow}$	0.305 0	0.182 8

文本检索模型 vs. 域检索模型 BM25F 优于其对应的纯文本模型 BM25，FSDM 优于其对应的纯文本模型 SDM。故很明显的结论是：将实体做不完全扁平化处理优于完全扁平化处理为纯文本。

域检索模型 vs. 扩展的域检索模型 在两个数据集上，不论只依赖于单一特征空间，还是依赖于双特征空间，采用排序学习算法来学习域权重的扩展域检索模型在大部分时候要优于单纯的域检索模型。至此，两者的性能对比验证了本文采用排序学习算法获得域检索模型权重的合理性。

文本检索模型 vs. D-L2RE D-L2RE 与文本检索模型中具有最优性能的模型相比，在 DBpedia 上 NDCG@10 相较于 SDM 提升 23.1%，在 OHSUMED 上 MAP 比 BM25 提升 17.5%。

域检索模型 vs. D-L2RE D-L2RE 与域检索模型中最佳性能的模型比，在 DBpedia 上比 FSDM 的 NDCG@5 提升 12.2%，在 OHSUMED 上比 BM25F 的 MAP 提升 3.0%。作为扩展域模型的代表 D-L2RE 优于域模型结果是显然的。

单特征空间实体排序学习 vs. D-L2RE 无论是在 DBpedia 上，还是在 OHSUMED 上，实体特征空间的实体排序学习算法 E-L2RE 性能优于词空间的实体排序学习算法。相比而言，在医学领域数据集上，E-L2RE 比 W-L2RE 优势更为明显，原因主要在于专业领域术语很多，在词空间中会把这些术语拆分成没有意义的词。但是查询文本通常较短，如果仅用实体特征建模会过于稀疏。因此，本文提出的双特征空间实体排序学习算法可以实现实体特征空间与词特征空间的优势互补，这一点在大部分的评价指标上可以体现出来。DBpedia 数据集上，D-L2RE 的 MAP 高于最优的单特征空间实体排序学习模型 E-L2RE 约 10%。OHSUMED 数据集上，D-L2RE 的 NDCG@5 性能比 E-L2RE 高约 2.2%。

纯文本检索并不能适用于实体检索任务。引入

实体特征空间对实体检索任务来说是必要的。本文提出的双特征空间实体排序学习算法在实体检索任务中性能优于其他实体检索模型。

3.3　特征选择

D-L2RE 包含实体与词两个特征空间。通过实验调研不同特征空间对 D-LRE 性能的影响。本文从特征选择的角度试图回答以下两个问题：①是否两个空间都在起作用？②哪个空间起的作用更大一些？

首先，对实体空间和词空间分别计算表 1 给出的所有 12 个特征与真实相关性标注的相关度，如用 NDCG@10 来计算，分别根据相关度对 12 个特征的重要程度排序。接着，对实体空间与词空间的维度分别取前 d_w 维与前 d_e 维构成双特征空间。最后，计算 D-L2RE 在不同维度特征空间中的性能对比，如图 2 所示。

OHSUMED 数据集上在图 2(a)与 2(b)上表现比较明显：右下角颜色最深表明实体特征与词特征维度相当的时候性能最优。这说明双特征空间实体排序学习算法 D-L2RE 的优越性是由两个空间同时起作用。

图 2(b)、2(c)与 2(d)中，右侧整体偏深。只要保证实体空间的特征维度大一些，即使词空间的特征维度很小，也能取得很好的效果。这说明在双特征空间中，实体特征空间的作用更大一些。

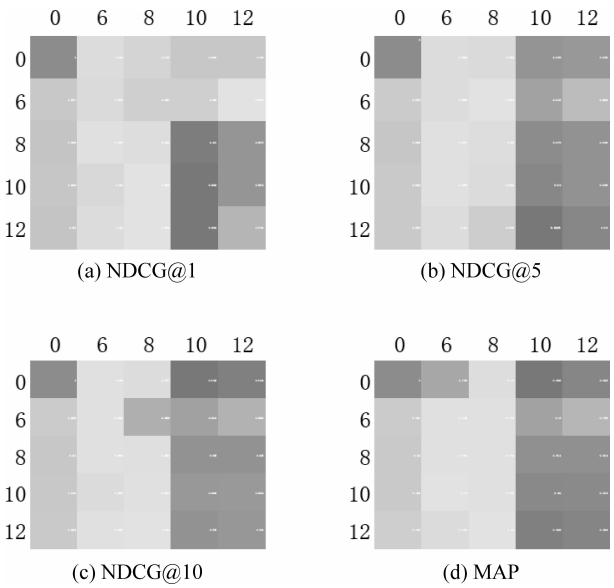

图 2　D-L2RE 在不同维度特征空间中的性能对比

OHSUMED 上双特征空间的不同维度(d_w,d_e)对 D-L2RE 性能的影响，其中由浅色至深色数值由大变小。

3.4　收敛分析

通过特征选择分析可知，虽然两个特征空间的融合可以带来检索性能的提升，但并不如所想的那样检索精度也随着特征的增加而增加。双特征空间的融合，一方面会使数据维度增加，另一方面会带来噪声。这些既会增加单步计算时间，也会影响模型收敛速度，使得训练时间变长。

Pairwise 排序学习算法 RankSVM 优化目标的上界是类似于 Kendall Tau① 的文档对的误分类率[30]。为了便于分析，本文展示了最优性能参数配置下的 W-L2RE($d_w=8$)，E-L2RE($d_e=6$)，以及 D-L2RE($d_w+d_e=14$)三个方法在 DBpedia 的训练集上，Kendall Tau 随迭代次数的增加而变化的情况，如图 3 所示。

可以看出，E-L2RE 收敛最快，约 3 000 次迭代后收敛。W-L2RE 与 D-L2RE 收敛速度相当，约 3 300 次迭代收敛。D-L2RE 稍微慢一些，约 3 400 次迭代收敛。本文提出的双特征空间融合方法，如式(3)所示，直接导致了数据特征维度升高，从而引起模型变得复杂。在同样的训练实例数与同样的训练算法的前提下，复杂模型收敛更慢一些。

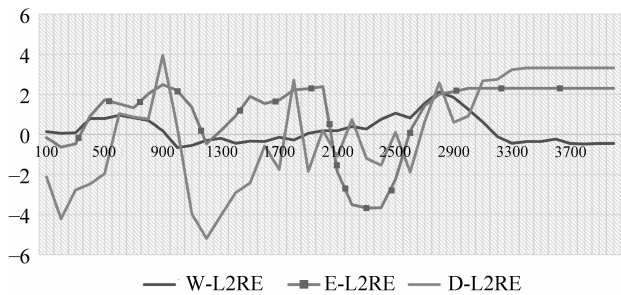

图 3　不同特征空间下的实体排序学习算法在 DBpedia 的训练集上的学习曲线

4　总结

本文提出了双特征空间的实体排序学习算法 D-L2RE 来融合实体特征空间与词特征空间。实体特征空间，可以表达出语义信息，以弥补词特征空间的不足，表达出语义信息。然而，实体特征空间的稀疏性造成的某些不确定性，需要词特征空间来弥补。最后本文在实体检索的标准数据集上验证了

① https://en.wikipedia.org/wiki/Kendall_rank_correlation_coefficient

D-L2RE的有效性。

下一步工作主要包括三个方面：①利用知识图谱中的关系数据来辅助检索性能的提升；②将双特征空间的思想从检索模型推广到匹配模型；③探索更为合理的双特征空间融合方法。

参考文献

[1] Pujara J, Singh S. Mining knowledge graphs from text [C]//Proceedings of the 11th ACM International Conference on Web Search and Data Mining. ACM, 2018: 789-790.

[2] Zhiltsov N, Kotov A, Nikolaev F. Fielded sequential dependence model for ad-hoc entity retrieval in the web of data [C]//Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2015: 253-262.

[3] Robertson S, Zaragoza H, Taylor M. Simple BM25 extension to multiple weighted fields[C]//Proceedings of the 13th ACM International Conference on Information and Knowledge Management. ACM, 2004: 42-49.

[4] Ogilvie P, Callan J, Callan J. Combining document representations for known-item search[C]//Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2003: 143-150.

[5] Wang Q, Mao Z, Wang B, et al. Knowledge graph embedding: A survey of approaches and applications [J]. IEEE Transactions on Knowledge and Data Engineering, 2017, 29(12): 2724-2743.

[6] Schütze H, Manning C D, Raghavan P. Introduction to information retrieval[C]//Proceedings of the International Communication of Association for Computing Machinery Conference. 2008: 260.

[7] Deerwester S, Dumais S T, Furnas G W, et al. Indexing by latent semantic analysis[J]. Journal of the American Society for Information Science, 1990, 41(6): 391-407.

[8] Robertson S E, Walker S. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval [C]//Proceedings of SIGIR'94. Springer, London, 1994: 232-241.

[9] Ponte J M, Croft W B. A language modeling approach to information retrieval[D]. University of Massachusetts at Amherst, 1998.

[10] Metzler D, Croft W B. A Markov random field model for term dependencies[C]//Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2005: 472-479.

[11] Joachims T. Optimizing search engines using clickthrough data [C]//Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2002: 133-142.

[12] Burges C, Shaked T, Renshaw E, et al. Learning to rank using gradient descent[C]//Proceedings of the 22nd International Conference on Machine Learning (ICML-05), 2005: 89-96.

[13] Cao Z, Qin T, Liu T, et al. Learning to rank: From pairwise approach to listwise approach[C]//Proceedings of the 24th International Conference on Machine Learning, 2007: 129-136.

[14] Burges C J C. From ranknet to lambdarank to lambdamart: An overview[J]. Learning, 2010, 11(23-581): 81-99.

[15] Huang P S, He X, Gao J, et al. Learning deep structured semantic models for web search using clickthrough data[C]//Proceedings of the 22nd ACM International Conference on Information & Knowledge Management. ACM, 2013: 2333-2338.

[16] Guo J, Fan Y, Ai Q, et al. A deep relevance matching model for ad-hoc retrieval[C]//Proceedings of the 25th ACM International Conference on Information and Knowledge Management. ACM, 2016: 55-64.

[17] Nikolaev F, Kotov A, Zhiltsov N. Parameterized fielded term dependence models for ad-hoc entity retrieval from knowledge graph[C]//Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2016: 435-444.

[18] Park D H, Liu M, Zhai C X, et al. Leveraging user reviews to improve accuracy for mobile app retrieval [C]//Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2015: 533-542.

[19] Xiong C, Power R, Callan J. Explicit semantic ranking for academic search via knowledge graph embedding[C]//Proceedings of the 26th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2017: 1271-1279.

[20] Xiong C, Callan J, Liu T Y. Word-entity duet representations for document ranking[C]//Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2017: 763-772.

[21] Jameel S, Bouraoui Z, Schockaert S. Member: Maxmargin based embeddings for entity retrieval[C]// Proceedings of the 40th International ACM SIGIR

Conference on Research and Development in Information Retrieval. ACM, 2017: 783-792.

[22] Meij E, Weerkamp W, De Rijke M. Adding semantics to microblog posts[C]//Proceedings of the 5th ACM International Conference on Web Search and Data Mining. ACM, 2012: 563-572.

[23] Ratinov L, Roth D, Downey D, et al. Local and global algorithms for disambiguation to wikipedia [C]//Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (Volume 1). Association for Computational Linguistics, 2011: 1375-1384.

[24] Ferragina P, Scaiella U. TAGME: On-the-fly annotation of short text fragments (by wikipedia entities) [C]//Proceedings of the 19th ACM International Conference on Information and Knowledge Management, 2010: 1625-1628.

[25] Cornolti M, Ferragina P, Ciaramita M. A framework for benchmarking entity-annotation systems [C]// Proceedings of the 22nd International Conference on World Wide Web. ACM, 2013: 249-260.

[26] Xiong C, Callan J, Liu T Y. Bag-of-entities representation for ranking[C]//Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval. ACM, 2016: 181-184.

[27] Qin T, Liu T Y, Xu J, et al. LETOR: A benchmark collection for research on learning to rank for information retrieval[J]. Information Retrieval, 2010, 13 (4): 346-374.

[28] Balog K, Neumayer R. A test collection for entity search in DBpedia[C]//Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2013: 737-740.

[29] Jarvelin K, Kekalainen J. Cumulated gain-based evaluation of IR techniques[J]. ACM Transactions on Information Systems, 2002, 20(4): 422-446.

[30] Lan Y, Niu S, Guo J, et al. Is top-k sufficient for ranking? [C]//Proceedings of the 22nd ACM International Conference on Information and Knowledge Management, 2013: 1261-1270.

赵以昕(1996—)，助理工程师，主要研究领域为数据挖掘、信息检索。
E-mail: yixin@iscas.ac.cn

纪春岩(1963—)，通信作者，博士，教授，主要研究领域为恶性血液病的发病机制及耐药机制。
E-mail: jichunyan@sdu.edu.cn

牛树梓(1985—)，博士，副研究员，主要研究领域为网络搜索与数据挖掘。
E-mail: shuzi@iscas.ac.cn