ISSN 1004-9037,CODEN SCYCE4
Journal of Data Acquisition and Processing Vol. 32,No. 3,May 2017,pp. 604－611
DOI:10.16337/j.1004-9037.2017.03.020
© 2017 by Journal of Data Acquisition and Processing
http://sjcj.nuaa.edu.cn
E-mail:sjcj@nuaa.edu.cn
Tel/Fax:＋86-025-84892742

基于词向量的实体链接方法

齐爱芹　徐蔚然

(北京邮电大学自动化学院,北京,100876)

摘　要: 实体链接任务主要包括命名实体识别、查询扩展、候选实体选择、特征抽取和排序。本文针对查询词的扩展,提出了一种基于词向量的扩展方法。该方法利用连续词袋(Continuous bag-of-words,CBOW)模型训练语料中词语的词向量,然后将距离查询词近的词作为扩展词。词向量从语料中挖掘出词与词之间的语义相关性是对基于规则的查询扩展方法的补充,以此来召回候选实体。在特征抽取时,把文档之间的潜在狄利克雷分布(Latent Dirichlet allocation, LDA)的主题相似性作为特征之一。在计算文档相似性时,不再以高频词作为向量的维度,而是以基于词向量的相关词作为向量维度,由此得到文档的语义相似性特征 。最后利用基于单文档方法的排序学习模型把查询词链接到相应的候选实体。实验结果表明利用该方法能使 F_1 值达到 0.71,具有较好的效果。

关键词: 实体链接;潜在狄利克雷分布;词向量;排序学习
中图分类号: TP391　　**文献标志码:** A

Method of Entity Linking Based on Word Embedding

Qi Aiqin, Xu Weiran

(Automation School, Beijing University of Posts and Telecommunications, Beijing, 100876, China)

Abstract: Entity linking includes entity discovery, query expansion, candidate generation, feature extraction and ranking. Here the query expansion method based on word embedding is proposed. Word embedding of words are trained by continuous bag-of-words (CBOW) model. Then the related words become the expansion words. The related words could make up the expansion based on rule. The related words could recall more and more candidate words simultaneously. In the feature extraction, the topic similarity between texts is extracted as the feature based on latent Dirichlet allocation(LDA). This paper extracts the synonyms based on word embedding as the dimension of text vector. Finally, learning to rank model is used to select the best candidate entity. The result shows that the method can ensure F_1 reaching 0.71, and be effective for entity linking.

Key words: entity linking; latent Dirichlet allocation (LDA); word embedding; learning to rank

引　言

网络的飞速发展给人们生活带来了便利,与此同时,引发的信息爆炸让人们很难精确地定位所求的

基金项目: 国家自然科学基金(61273217)资助项目;国家高等学校博士学科点专项科研基金(20130005110004)资助项目。
收稿日期: 2015-10-07;**修订日期:** 2015-11-02

信息。各种搜索引擎的出现,如百度、搜狗、360 搜索和谷歌等为用户提供检索服务,将用户查询词的相关信息反馈给用户。由于自然语言的歧义性,例如,“苹果”既可以指水果中的苹果,也可以指生产电子产品的“苹果”公司,对实体的语义进行消歧成了查询的关键问题。基于文本解析会议(Text analysis conference,TAC)的实体链接(Entity linking)任务应运而生。实体链接任务是指抽取文档集中指定类型的命名实体,包括人名、地名和组织机构名,并把其链接到知识库(Knowledge base,KB)的过程。KB 是 2008 年的维基百科快照,是一个半结构化的知识库,每一个词条在 KB 中都是一个 KB 节点,对应一个唯一的 ID。如何根据给定的文档语境把实体链接到 KB 中是本任务的关键问题。本文提出了结合语义进行查询扩展以及基于语义的特征抽取方式。针对查询词的扩展,利用词向量的扩展方法,用神经网络去训练语料中词语的词向量。然后将距离查询词相近的词作为扩展词,词向量从语料中挖掘出词与词之间的语义相关性,此方法充分考虑了语义相关性,能把同义词,共指关系的词召回。此方法可以召回更多的相关候选实体,补充了基于规则的模糊匹配的查询扩展。在特征抽取时,充分考虑文档的主题,把文档之间的潜在狄利克雷分布(Latent Dirichlet allocation,LDA)的主题相似性作为特征之一。在计算文档相似性时,不再以高频词作为文本向量,而是以基于词向量的相关词作为向量维度,由此得到文档的基于语义的相似性特征。基于词向量的相关词在语义上比高频词更能代表文本。最后使用基于单文档的排序学习模型把查询词链接到候选实体上。本文以 2014 年的知识库扩充(Knowledge base population,kBP)评测数据为实验数据,结果显示该方法有较好的效果。

目前实体链接的方法主要有分类法、概率主题方法、Graph 方法和排序法。分类法[1]把每个候选实体看作一个类别,每一个查询词就是一个待分类项,抽取特征后根据 SVM 进行分类,此方法是哈尔滨工业大学在 2012 年 TAC 的实体链接任务中所采用的方法。但该方法没有考虑文档中的语义信息,只是根据词的共现来进行分类,并且训练数据少、噪声大。概率主题方法[2]挖掘隐藏在文本之间的主题关系,来衡量文本之间的相似性。此方法只是单纯地根据上、下文语义来进行实体链接,没有充分利用维基百科的结构化信息,准确率不是很高。Graph 方法[3]基于文本中实体和维基百科的特点,构造语义网络,通过网络中节点的距离、出度和入度等作为特征来设计相似度衡量指标,从而实现语义消歧。虽然这种方法有较好的链接效果,然而也存在一些问题,当数据规模比较大时,网络图的存储、训练和计算就会比较繁琐复杂。传统的排序法即向量空间模型全称(Vector space model,VSM)[4]抽取实体的上下文作为词袋,然后根据词频-逆向文档频(Term frequency-inverse document frequency, TF-IDF)中向量空间(Vector space model)把上、下文表示成文本向量,计算余弦相似性。VSM 模型是最传统的命名实体消歧方法的模型,这种方法认为同一个命名实体的上、下文更相似。Bagga 和 Baldwin[5]使用“词袋”模型的方法计算文章相似度,来判断某两篇文档中出现的相同命名指称项是否对应同一命名实体,“词袋”模型只考虑了上、下文信息,没有挖掘深层次的语义关系,也没有加入其他重要的特征。基于以上分析,本文提出融合结构特征和语义特征的排序学习模型进行实体链接。本文充分利用了维基百科的半结构化信息,抽取了页面的 Infobox 信息特征,同时考虑语义特征,抽取文档之间的 LDA 主题相似度,计算基于词向量的语义相似度特征,使用单文档方法构建排序学习模型,把此问题转变成回归问题,进行排序。实验结果显示本方法在 2014 年 TAC 评测中对实体链接任务有较好的效果。现有的命名实体识别方法主要有基于词典和规则的算法,基于隐马尔科夫模型的方法和基于条件随机场(Condition random field,CRF)的方法。基于词典和规则的算法利用现有的词典和人工规则进行命名实体识别,但是这种方法存在两大问题:(1)词典的构建,词典是否完全决定了实体识别的性能。(2)实体识别规则的确定,规则之间的冲突难以避免,并且迁移性较差。基于隐马尔科夫模型的命名实体识别采用 Viterbi 算法对初始观察序列标注,求出最佳的状态序列,用 K 均值方法进行估计,并将估计结果使用线性插值法进行平滑。基于 CRF 的命名实体识别,是一种用于标注和切分有序数据的条件概率模型,集合了最大熵模型和隐马尔可夫(Hidden Markov model,HMM)模型的特点。CRF 可以看作是一种无向图模型,考察

给定输入序列的标注序列的条件概率。

查询扩展的方法主要有基于统计的查询扩展和基于语义词典的查询扩展。基于统计的查询扩展是根据语料中词的上、下文相同词的共现概率来进行扩展。基于统计的方式在训练语料过大时容易导致语义漂移。基于语义词典的查询扩展是根据人工标注的语义词典进行扩展。主要的语义词典有 WordNet、HowNet 和同义词林等。基于词典的扩展方法过多地依赖现有词典,对于词典中未登录词达不到好的扩展效果。于是本文采用基于规则和词向量相结合的方式进行查询扩展,能够达到较好的扩展效果。

1 系统描述

基于任务分析,实体链接主要解决的关键性问题有:命名实体识别、查询扩展、候选实体生成、特征抽取和排序。具体的系统流程图如图 1 所示。

1.1 基于条件随机场的命名实体识别

根据本任务要求,命名实体识别识别出人名、地名和组织结构名这 3 种类型的实体。本文使用 CRF 的方法来解决任务中的命名实体识别任务,训练数据是评测任务给定的训练集,利用 Stanford CoreNLP NER 得到数据的句法树。对于训练数据的标记采用 B,E,I 和 O 来标志,B 表示实体的开始,E 表示实体的结束,O 表示非实体,I 表示实体内部。选择的特征包括词语、词性、词在句法树中的父节点以及和父节点的关系。采用的模板参数如表 1 所示。表 1 中 T * *:%×[#.#]中的 T 为模板类型,两个"#"分别为相应的行偏移和列偏移。

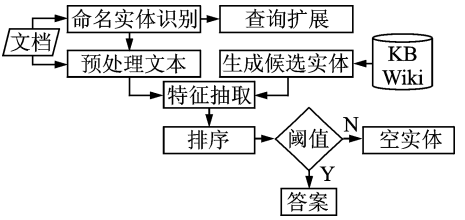

图 1 系统流程图

Fig. 1 System flow chart

表 1 CRF 的模板参数

Tab. 1 Template parameter of CRF

# Unigram		
U00:%×[-2,0]	U10:%×[-2,1]	U17:%×[0,1]/%×[1,1]
U01:%×[-1,0]	U11:%×[-1,1]	U18:%×[1,1]/%×[2,1]
U02:%×[0,0]	U12:%×[0,1]	U20:%×[-,1]/%×[-,1]/%×[0,1]
U03:%×[1,0]	U13:%×[1,1]	U21:%×[-,1]/%×[0,1]/%×[1,1]
U04:%×[2,0]	U14:%×[2,1]	U22:%×[0,1]/%×[1,1]/%×[2,1]
U05:%×[-,0]/%×[0,0]	U15:%×[-2,1]/%×[-1,1]	U23:%×[0,1]
U06:%×[0,0]/%×[1,0]	U16:%×[-1,1]/%×[0,1]	# Bigram

1.2 查询扩展

查询扩展在实体链接的过程中是较为关键的一步,它对提高实体链接的准确率和召回率具有重要的影响。在候选实体选择模块中,查询词往往是缩写,于是对缩写进行扩展至关重要。比如,在维基百科中有超过几十个条目的缩写都是"ABC",但如果将"ABC"扩展为"American Broadcasting Company",这样在 KB 知识库中就可以准确地召回候选实体,而过滤其他不相关实体。本实验使用基于规则和基于词向量的方法进行查询扩展来提高准确率和召回率。

1.2.1 基于规则查询扩展

在基于规则进行扩展时,主要使用支撑文档进行扩展,支撑文档即官方给定的出现这个查询词的文档,本文扩展时使用的规则如下:

(1) 对于人名,若文档中有全称,就把查询词扩展成全称,如"Brown"扩展成"John Brown"。

(2) 对于大写缩写,借助 CRF 抽取的命名实体在文档中让缩写扩展成全称。比如 IBM 在文档中表

示魔术协会,借助停用词扩展成"International Brotherhood of Magicians"。

(3) 对于地名的缩写,根据地名、别名缩写词典进行扩展,如"US"扩展成"United States"。

1.2.2 基于词向量的查询扩展

对查询词进行扩展往往是为了提高准确率,把一个查询词扩展成一个比较精确的查询词,如"Lucy"扩展成"Lucy Walsh",但由于支撑文档只是出现查询词的背景,或者在文档中查询词只出现一次,于是借用支撑文档进行基于规则的查询扩展有可能变得无效。为了能召回比较精确的候选实体,于是本文借用 Wiki 的外部数据进行查询扩展。比如"Lucy"若不能进行基于规则的扩展,则基于词向量的扩展方法让返回相似度达到 0.5 以上的实体作为扩展词,此扩展词就作为候选实体来进行链接,以"Detroit"为例,基于词向量的扩展结果有实体"Detroit Red Wings","USS Detroit"等,然后在 KB 中索引这些扩展实体的页面作为候选实体。基于词向量的查询扩展根据词的上、下文语义进行扩展,是一个基于统计的无监督训练方式,此方法认为相近的词在语义上距离更相近。Word2vec[6] 是 Google 公司在 2013 年提出的一款训练词向量的工具,准确性高。Word2vec 是一个深度学习模型,同神经网络语言模型相似。Word2vec 以大量文本训练语料作为输入,可以将每个词特征化为一个 K 维的实值向量,在该向量上进行相似度计算将能挖掘出相似词。Word2vec 包括 CBOW 中袋和 Skip gram model 两种模型,本文使用 CBOW 模型。CBOW 模型与传统的前向神经网络语言模型类似,不同点在于 :(1) CBOW 预测语句中间的词,而不是最后一个词。(2) CBOW 去掉了模型计算中最耗时的非线性隐层并且对所有词而言隐层相同,从而有效地提高了词向量的训练速度。CBOW 如图 2 所示。训练词向量的参数设置为/word2vec-train kb. txt-output kb. vector-cbow 1-size 100-window 5-negative 0-hs 1-sample le-3-threads 12-binary 1。训练词向量时使用 KB,使用 CBOW 模型。使用二进制格式来存储词向量结果,参数如表 2 所示。

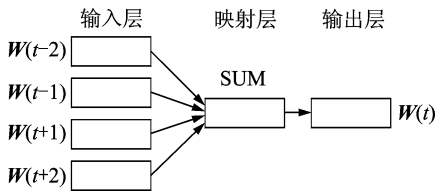

图 2 CBOW 模型

Fig. 2 CBOW model

表 2 词向量参数

Tab. 2 Parameter of word2vec

参数	含义
-train	训练语料
-output	模型的输出文件
-cbow	"1"代表使用 CBOW 模型,"0"则是不使用 CBOW。
-size	设置输出词向量的维度为 100。
-window	设置训练神经网络语言模型的词窗大小为 5,即认为当前单词只与其、前后 5 个单词相关。
-negative	表示不使用负采样方法。
-hs	表示使用层次 Softmax(Hierarchical Softmax)方法。
-sample	设置采样频率阈值,若一个单词被采样,那么意味着它在训练预料中出现的频率值大于设定的阈值。
-threads	训练模型的线程数。
-binary	"1"表示采用二进制存储计算结果。

1.3 候选实体的选择

为了能快速返回候选实体,本文使用 Indri 对 KB 建立倒排索引,Indri 是一个完整的开源搜索引擎,由卡耐基-梅隆大学 Lemur 项目组维护并持续开发。为了能最大程度地召回正确实体,本文进行索引的查询词是由规则方法进行扩展的查询词和由词向量进行扩展形成的查询词,使用模糊匹配的方式返回候选实体。例如,"Lucy"的候选实体为"Lucy Walsh""I_Love_Lucy""Lucy_Coe"等。由于 KB 页面中的信息只有别名,标题,文本信息。根据文献[7],外部数据 Wikipedia 能提供更多的特征。于是本文对

20140203 的"Enwiki"进行解析，对于返回候选实体的标题，在"Enwiki"中抽取其中的"Category_inlinks" "Category_pages""MetaData""Page_inlinks" "Page_outlinks" "Page""Category_outlinks""Category"等信息并存入 11 个 Mysql 表中。KB 页面和 Wikipedia 页面中标题都是唯一的标识，对于每一个对应的 KB 页面，都在 Wikipedia 中搜索到相应的页面来扩展候选实体的特征。

1.4 特征抽取

在训练排序模型时，要抽取查询词与候选实体的特征，特征的选择直接影响排序模型的效果。无论是清华大学[8]或是中国科学院计算技术研究所[9]的 2014 年评测、抽取的特征都包括名称相似性、实体流行度、上下文相似性、类型和缩写。本文认为基于语义的主题信息仍是一个关键因素。对于不同主题的文章，上下文往往是不同的。本文抽取了 LDA 的主题特征以及基于词向量的文本相似性特征。

(1) 实体之间的信息。实验结果显示，查询词和候选实体标题的相关性对实验结果有着重要的影响。例如扩展的"Lucy Walsh"与 KB 中的"Lucy Walsh"完全一致，结果显示其是正确的链接 KB 节点。于是查询词与标题的相似性是特征之一，本文使用编辑相似性来度量其相似性，则

$$S=1-\frac{L'}{\max(L_1,L_2)} \tag{1}$$

式中：L' 为最小编辑次数，L_1，L_2 分别为字符串的长度。

(2) Wiki 中的实体 E。信息窗 Infobox 中的信息包括本地名、法定名、昵称以及上下文中的实体，上下文为 KB 中的文本内容。E_i 为

$$E_i=\begin{cases}1 & \text{如果第 } i \text{ 个实体出现在实体 } e \text{ 的上下文中}\\ 0 & \text{其他}\end{cases} \tag{2}$$

(3) 实体的类别特征 C。类别标签反应的是一个实体的类别，若两个实体表示同一实体，那么其类别一致。本文抽取支撑文档中词的所有能确定的类别作为查询词的类别以及候选实体的类别，C_i 对其进行评估，C_i 为

$$C_i=\begin{cases}1 & \text{如果实体的类目和查询词的一致}\\ 0 & \text{其他}\end{cases} \tag{3}$$

(4) 类型信息 t。类型信息指一个实体的类型，对于命名实体识别出的实体类型包含 PER，GPE，ORG。而 KB 中实体也有相应的实体类型，如 UKN，PER，GPE。UKN 为不能确定的类型。若实体的类型不是 UKN 并且实体的类型和查询词的类型不一致则 t 为 0。否则 t 为 1。t 的定义为

$$t=\begin{cases}0 & \text{如果实体的类型不是未知，}\\ & \text{同时查询词和实体有相同的类型}\\ 1 & \text{其他}\end{cases} \tag{4}$$

(5) 基于主题模型的文档相似度。LDA[10]模型是 Blei 提出的一种对文档集建模的概率主题模型。传统判断两个文档相似性的方法是通过统计两篇文档共同出现的单词，基于 TF-IDF 的相似性计算方法，这种方法没有考虑到文档的语义相关性，而 LDA 恰好能表示两篇文档的主题相似性。LDA 模型认为一篇文章都是以一定概率选择了某个主题，并从这个主题中以一定概率选择某个词语，这样一个过程得到的一篇文档。LDA 的原理可以表示为

$$P(\text{词}\mid\text{主题})=\sum_{\text{主题}}P(\text{词}\mid\text{主题})\times P(\text{主题}\mid\text{文档}) \tag{5}$$

给定一系列文档，计算各个文档中每个单词的词频就可以得到“文档-词语”矩阵。主题模型就是通过这个“文档-词语”矩阵进行训练，学习出“词-主题”矩阵和“主题-文档”矩阵。LDA 的工作原理可以表示为 LDA 模型认为文档的主题分布和主题的词分布属于 LDA 分布，文档以多项分布的概率选择一个主题，主题以多项分布的概率选择一个词。然后用主题和词的联合分布来近似估计主题的后验分布。

最后训练文档的主题分布,用 KL 散度来计算文档之间的主题相似性。KL 距离也即相对熵,表示两个概率分布的距离。相似度度量标准 KL 距离为

$$D_{\mathrm{KL}}(p,q)=\sum_{j=1}^{T} p_j \ln \frac{p_j}{q_j} \tag{6}$$

式中:p ,q 分别为两个概率分布。

(6) 基于词向量的文本相似性。由查询扩展中对词向量的概述所知,通过词向量进行查询扩展时,对文档中的词利用神经网络进行词向量训练,然后根据余弦距离得到一个词的近义词或者相关词。传统的度量文档之间的相似性是基于高频词的,但是实体链接的特殊之处在于,支撑文档只是查询词出现的语料,文档并不能真正地解释查询词的含义,于是文档的高频词并不一定是经常和查询词一起出现的高频词。同时支撑文档中的高频词也不能很好地表达查询词的语义信息。为了能用其他语义共现词来表示查询词的上、下文,本文使用基于词向量的近义词作为查询词的文本向量。词向量是基于上、下文语义来训练的,于是得出的相关词更能代表词的上、下文语境。传统的基于词频的文本相似性往往导致稀疏矩阵的计算,造成维度灾难。为了避免这个问题,本文使用的维度是 50 维,由得到的相似值可以推出,排名 100 的词相似性达到 0.2 左右,于是不再作为相关词对待。同时在训练词向量时也要考虑歧义词问题,比如"Lucy"的近义词可能包含"Lucy Walsh"的近义词,也可能包含"I Love Lucy"的近义词。为了防止此问题引入的噪声,在基于规则的方法进行扩展的前提下,对扩展词使用基于词向量的相近词文本向量化,比如"Lucy"扩展成"Lucy Walsh",文本向量化时使用基于词向量的近义词。为了防止噪声的出现,假如不能扩展"Lucy",仍旧使用支撑文档中的实体作为上下文。对于由相关词作为维度构成的文本向量,使用余弦距离来度量文本之间的语义相似性。查询词"Apple Inc"的距离最近的几个词如表 3 所示。余弦距离用 Sim 表示为

$$\mathrm{Sim}=\frac{\sum_{i=1}^{n}(A_i\times B_i)}{\sqrt{\sum_{i=1}^{n}A_i^2}\sqrt{\sum_{i=1}^{n}B_i^2}} \tag{7}$$

表 3 "Apple Inc"基于词向量的相近词

Tab. 3 Related words to "Apple Inc" based on word2vec

词	余弦距离	词	余弦距离
Software	0.605 103	IBM	0.521 855
Computer	0.573 526	Microsoft	0.514 399
Apple's	0.560 684	IPod	0.504 693
Desktop	0.537 930	Intel	0.496 765
HP	0.525 317		

2 基于单文档排序的 LTR 模型

抽取的特征能否提高实体链接的准确率是排序学习(Learning to rank,LTR)模型[11,12]要解决的核心问题。LTR 是一种监督学习的排序方法,已被广泛应用到文本挖掘的很多领域。LTR 融合多种特征对候选结果进行排序。LTR 有 3 类方法:单文档方法、文档对方法和文档列表方法,本文使用单文档方法来进行学习排序。单文档方法处理对象单一文档,将文档转化为特征向量后,主要是将排序问题转化为机器学习中常规的分类或回归问题。当模型参数学习完毕后,就可利用模型进行相关性判断。对新的查询和文档,通过模型的排序函数可以得到一个数值,利用该数值即可对文档进行排序。根据对文档的分析以及 2013 年 KBP 实验室结果的训练,得出查询词与候选实体的排序函数为

$$\mathrm{Score}(q,e_i)=\frac{\left(\varphi_1 S_i+\varphi_2\frac{E_i}{\sum_{j=1}^{n}E_j}+\varphi_3\frac{C_i}{\sum_{j=1}^{n}C_j}+\varphi_4\ \mathrm{sim}_i\right)}{\varphi_1+\varphi_2+\varphi_3+\varphi_4}\times t\times D_{KL} \tag{8}$$

式中：S_i 为查询词与实体的编辑相似性；E_i 为实体页面内实体在查询词的支撑文档中出现的次数；$\sum_{j=1}^{n} E_j$ 为总的候选实体的页面内实体出现的总次数；C_i 为实体内页面类别在查询词的支撑实体的类别中出现的次数；$\sum_{j=1}^{n} C_j$ 为总的候选实体的类别在查询词的支撑实体类别中出现的总次数；φ_1，φ_2，φ_3，φ_4 分别为各个特征所占的权重；sim_i 为支撑文档与候选实体文档的相似性；t 为类型信息，取值为 0 或 1；D_{KL} 为文档之间的主题相似度。最后，根据排序函数，若 Score 小于 0.6，则设定为空实体。根据任务要求，对空的实体需要再进行聚类，使用杰卡德相似系数(Jaccard similarity coefficient，JSC)进行度量。即根据各个空实体支撑文档中共现的词个数进行聚类，大于某一阈值，就聚为一类。杰卡德系数为

$$d_J(A,B)=1-J(A,B)=\frac{|A\cup B|-|A\cap B|}{|A\cup B|} \tag{9}$$

3 实验设计与结果分析

本次实验使用 2014 年 TAC 评测中实体链接任务提供的 138 篇文档作为测试数据，共抽取了 5 234 个命名实体。用 Indri 对 KB 进行索引，设置的 N 为 50，即对每个查询词，返回前 50 个候选实体，然后用 Java 解析维基百科包(Java Wikipedia library，JWPL)解析 Enwiki-20140203-pages-articles.xml 文档，返回其重定向页、消歧页以及页面内其他信息。抽取实体的特征，利用 LTR 模型进行实体链接。实验结果如表 4 所示。实验结果的衡量使用 F_1 值。

准确率为

$$P=\frac{\text{链接正确的查询词数}}{\text{链接的查询词数}} \tag{10}$$

召回率为

$$R=\frac{\text{链接正确的查询词数}}{\text{答案中链接的查询词数}} \tag{11}$$

$$F_1=\frac{(\alpha+1)P\times R}{\alpha\times P+R} \tag{12}$$

表 4 实验结果

Tab. 4 Experimental results

评价参数	实验 1	实验 2	实验 3	实验 4
B^3+F_1(All-5234)	0.523	0.545	0.634	0.710
B^3+F_1 (in KB-2817)	0.519	0.411	0.570	0.690
B^3+F_1 (not in KB-2417)	0.532	0.696	0.710	0.732
B^3+F_1 (NW docs-1575)	0.541	0.560	0.60	0.710
B^3+F_1 (WB docs-1743)	0.490	0.558	0.569	0.650
B^3+F_1 (DF docs-1916)	0.539	0.511	0.723	0.760
B^3+F_1 (PER-3162)	0.562	0.579	0.710	0.770
B^3+F_1 (ORG-1007)	0.490	0.568	0.535	0.640
B^3+F_1 (GPE-1065)	0.432	0.417	0.485	0.590

本文进行了 4 组实验，实验的评价标准是 B^3+F_1，并且对结果在不同的方面进行评估，包括对在 KB 中和不在 KB 中的查询词进行评估。同时给定的文档还包括新闻、微博、网页(NW，WB，DF 三种类型)和对查询词的不同类型进行评估，包括 PER，ORG，GPE。实验 1 是基于 SVM[13] 的分类模型，使用实体的上、下文信息作为特征，每一个候选实体为一类，然后使用径向基核函数(Radial base-function，RBF)进行分类。从实验结果可以看出只是简单地使用上、下文信息进行 SVM 分类，效果不如其他方法。实验 2 使用候选实体的标题以及 KB 的别名信息来进行实体链接的向量空间模型。实验结果虽然不是很理想，但是从实

验中也可以看出,KB 节点的标题信息和别名信息对实体链接有着重要的作用。实验 3 抽取了半结构化的知识库 Wikipedia 进行解析,抽取其中的类别、别名、类型和文档之间的 LDA 主题相似性进行实验,与实验 2 对比显示加入了外部数据库特征以及主题相似性特征的实验在 PER、GPE 类型的实体方面效果好于实验 2,有着较好的作用,但是在 ORG 方面略低。在总体的 KB 内和不在 KB 内方面都有较好的效果,可以提高实验的 F_1 值。实验 4 是本文的 LTR 模型,本实验最主要的不同在于除了充分利用标题、类别、别名、上下文和 LDA 等信息之外,加入了基于词向量的文本相似性特征。与前两个实验结果相比,实验 4 在 GPE,ORG 类型的实体方法有着显著的提高,并且此方法对 All-Query 的链接也比其他方法有效。

4 结束语

本文是基于 2014 年 TAC 评测中实体链接任务进行的研究,在查询扩展中使用基于规则的方法及基于词向量的查询扩展方法。在特征抽取中,提出了融合 LDA 主题特征与基于词向量的文档相似性特征的排序学习模型,通过实验对比,显示此方法对实体链接有着较好的效果。但是此实验仍旧有不足之处,词向量的训练和语料的长短有很大关系。对于较短的文本,通过词向量很难找到与其相近的词。本实验训练词向量的窗口是 5,应该进一步调整词窗大小,进行实验来提高本实验的 F_1 值。

参考文献:

[1] Jian Xu, Qin Lu, Jie Liu, et al. NLPComp in TAC 2012 entity linking and slot-filling[EB/OL]. https://tac.nist.gov//publications/2012/participant.papers/NLPComp.proceedings.pdf,2012-12-25, 2015-10-02.

[2] 怀宝兴,宝腾飞,祝恒书,等. 一种基于概率主题模型的命名实体链接方法[J]. 软件学报,2014,25(9):2076-2087.
Huai Baoxing, Bao Tengfei, Zhu Hengshu, et al. Topic modeling approach to named entity linking[J]. Journal of Software, 2014,25(9):2076-2087.

[3] Xian Peihan, Sun Le, Zhao Jun. Collective entity linking in web text: A graph-based method[C]// Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval. [S. l.]: ACM, 2011:765-774.

[4] 郭庆琳,李艳梅,唐琦. 基于 VSM 的文本相似度计算的研究[J]. 计算机应用研究,2008,25(11):3256-3258.
Guo Qinglin, Li Yanmei, Tang Qi. Similarity computing of documents based on VSM[J]. Application Research of Computers, 2008,25(11):3256-3258.

[5] Bagga A, Baldwin B. Entity-based crossdocument coreferencing using the vector space model[EB/OL]. http://www.aclweb.org/anthology/P98-1012,1998-12-20/2015-10-02.

[6] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[EB/OL]. https://arxiv.org/abs/1301.3781,2013-12-10/2015-10-02.

[7] Cucerzan S. Large-scale named entity disambiguation based on Wikipedia data[EB/OL]. https://pdfs.semanticscholar.org/1c90/9ac1c331c0c246a88da047cbdcca9ec9b7e7.pdf,2007-12-15/2015-10-02

[8] Zhao Gang, Lü Ping, Xu Ruochen, et al. MSIIPL_THU at TAC 2014 entity discovery and linking track[EB/OL]. https://tac.nist.gov/protected/2014/TAC2014-workshop-notebook/participant.papers/TAC2014.MSIIPL_THU.notebook.pdf, 2014-12-25/2015-10-02.

[9] Hailun L Zeya Z Yantao. OpenKN at TAC KBP[EB/OL]. https://tac.nist.gov/publications/2015/participant.papers/TAC2015.ICTCAS_OKN.proceedings.pdf,2015-05-10/2015-10-02.

[10] 王振振,何明,杜永萍. 基于 LDA 主题模型的文本相似度计算[J]. 计算机科学,2013,40(12):229-232.
Wang Zhenzhen, He Ming, Du Yongping. Text similarity computing based on topic model LDA[J]. Computer Science, 2013, 40(12):229-232.

[11] Liu T Y. Learning to rank for information retrieval[J]. Foundations and Trends in Information Retrieval, 2009, 3(3): 225-331.

[12] Hang L. A short introduction to learning to rank[J]. IEICE Transactions on Information and Systems, 2011, 94(10): 1854-1862.

[13] 刘绍毓,周杰,李弼程,等. 基于多分类 SVM-KNN 的实体关系抽取[J]. 数据采集与处理,2015,30(1):202-210.
Liu Shaoyu, Zhou Jie, Li Bicheng, et al. Entity relation extraction method based on multi-classifier[J]. Journal of Data Acquisition and Processing, 2015,30(1):202-210.

作者简介:

齐爱芹(1989-),女,硕士研究生,研究方向:机器学习,E-mail:qiaiqin123@126.com。

徐蔚然(1975-),男,副教授,研究方向:模式识别。