一种基于语义关系图的词语语义相关度计算模型

张仰森 [1] 郑 佳 [1] 李佳媛 [1]

摘 要 词语的语义计算是自然语言处理领域的重要问题之一, 目前的研究主要集中在词语语义的相似度计算方面, 对词语语义的相关度计算方法研究不够. 为此, 本文提出了一种基于语义词典和语料库相结合的词语语义相关度计算模型. 首先, 以 HowNet 和大规模语料库为基础, 制定了相关的语义关系提取规则, 抽取了大量的语义依存关系; 然后, 以语义关系三元组为存储形式, 构建了语义关系图; 最后, 采用图论的相关理论, 对语义关系图中的语义关系进行处理, 设计了一个基于语义关系图的词语语义相关度计算模型. 实验结果表明, 本文提出的模型在词语语义相关度计算方面具有较好的效果, 在 WordSimilarity-353 数据集上的斯皮尔曼等级相关系数达到了 0.5358, 显著地提升了中文词语语义相关度的计算效果.

关键词 语义相关度, 语义关系图, HowNet, 依存语义关系, 语义相似度

引用格式 张仰森, 郑佳, 李佳媛. 一种基于语义关系图的词语语义相关度计算模型. 自动化学报, 2018, **44**(1): 87−98

DOI 10.16383/j.aas.2018.c170002

A Model for Calculating Semantic Relatedness of Words Considering Semantic Relationship Graph

ZHANG Yang-Sen[1] ZHENG Jia[1] LI Jia-Yuan[1]

Abstract Word semantic computation is one of the important issues in nature language processing. Current studies usually focus on semantic similarity computation of words, not paying enough attention to the semantic relatedness computation. For this reason, we present a word semantic relatedness calculation model based on semantic dictionary and corpus. First of all, the semantic extraction rules are formulated with "HowNet" and corpus, and a large number of semantic dependency relations are extracted based on these rules. Then, a semantic relationship graph is constructed by storing the semantic relationship triplet tuple. At last, graph theory is used to process the semantic relation in the semantic relationship graph and a semantic relatedness calculation model is designed by means of the semantic relationship graph. Experimental results show that this method has a better performance in word semantic relatedness computation, the Spearman rank correlation on the WordSimilarity-353 dataset being up to 0.5358, a significant efficiency improvement of semantic relatedness computation of Chinese words.

Key words Semantic relatedness, semantic relationship graph, HowNet, dependency semantic relation, semantic similarity

Citation Zhang Yang-Sen, Zheng Jia, Li Jia-Yuan. A model for calculating semantic relatedness of words considering semantic relationship graph. *Acta Automatica Sinica*, 2018, **44**(1): 87−98

从语义学的角度来看, 词语语义计算可以在词语表达的词义之间进行定义, 也可以在整个文本之中进行定义[1], 其表现形式主要有两种: 词语语义相关度和语义相似度. 词语语义相关度和相似度是两个不同的概念, 但两者之间又有着紧密的联系. 语义相关度反映的是两个词语相互关联的程度, 指的是词语之间的组合特点, 即看到一个词, 会自然而然地联想到另一个语义相关的词, 它可以用这两个词语在同一个语境中共现的可能性来衡量. 而语义相似度是指两个词语的相似程度, 通常指两个词语的语义本身具有某些相似的特性, 相似度反映的是词语之间的聚合特点, 即一个词可以用另一个语义相似的词替换. 就它们所表示的范畴来说, 语义相关度是更一般的概念, 而语义相似度是语义相关度的一种特例, 也就是, 语义相关度包含了语义相似度的概念.

Resnik[2] 用 "轿车"、"汽油"、"自行车" 的例子生动形象地解释了两者之间的区别: "轿车依赖于汽油作为燃料, 显然它们之间的相关性比轿车与自行车更为紧密, 但人们却普遍认为轿车与自行车之间的相似性大于轿车与汽油". 这个例子表明, 相关性不能等同于相似性. 即使轿车与汽油是紧密相关的,

收稿日期 2017-01-03 录用日期 2017-02-15
Manuscript received January 3, 2017; accepted February 15, 2017
国家自然科学基金 (61370139, 61602044) 资助
Supported by National Natural Science Foundation of China (61370139, 61602044)
本文责任编委 张民
Recommended by Associate Editor ZHANG Min
1. 北京信息科技大学智能信息处理研究所 北京 100101
1. Institute of Intelligent Information Processing, Beijing Information Science and Technology University, Beijing 100101

但由于这两者之间没有共同的特性, 人们不会认为它们是相似的. 而轿车和自行车都是交通工具, 都有轮子且可以载人, 因此它们是相似并且相关的. 再比如 "微软公司" 和 "比尔 · 盖茨" 是两个相关的词语, 比尔 · 盖茨是微软公司的创始人之一, 而且曾担任微软公司的 CEO, 提及微软公司, 我们可以很自然的联想到比尔 · 盖茨; 但他们并不是相似的, 并且它们也不能相互替换, 例如: "微软公司是一家具有创造力的公司", 如果将 "微软公司" 替换为 "比尔 · 盖茨" 将会出现错误, 而 "谷歌公司" 和 "微软公司" 是相似的词, 它们都是公司, 是可以相互替换的, 而且它们也是语义相关的. 由于词语相关度包含了相似度, 因此, 在评价词语相似度和相关度的时候, 可以把词语相似度作为相关度评价的一个维度, 也就是说, 如果词语间的语义越相似, 那么, 在一定程度上, 它们的相关度也越大, 相似度的大小在一定程度上影响着相关度的度量.

词语的语义相关度计算是许多自然语言处理任务的基础, 主要探索词语之间的相关程度. 在信息检索[3−4]、自动问答[5]、事件抽取[6−7]、词义消歧[8]、社会计算[9] 等自然语言处理的应用领域研究中, 词语的语义相关度计算都扮演着非常重要的角色. 本文旨在研究如何进行词语之间的相关度计算.

1 研究现状分析

目前, 针对词语语义相关度的评价, 已经提出了很多卓有成效的方法, 归纳起来主要分为两类: 基于语义词典的方法和基于统计的方法.

基于语义词典的方法主要是利用现有语义词典中的各种概念以及概念与概念之间的关系来度量词语的语义相关度. 英语的语义词典以 WordNet 为代表, Budanitsky 等[10] 总结了 5 种利用 WordNet 词典计算词语的语义相关度的方法, 并对它们的性能进行了比较. Taieb 等[11] 提出了一种新的 Information content (IC) 计算方法, 并在此基础上, 将 IC 融入到 WordNet 的分类系统中, 构建了词语语义相关度的计算模型. 而在中文中使用最多的语义词典是 HowNet, 其最早被引入语义计算的是刘群等[12], 他们在研究义原、集合和特征结构的相似度计算方法的基础上, 提出了利用 HowNet 进行词语语义相似度的计算算法. Zhang[13] 使用 HowNet 作为语义知识, 计算词语之间的语义相关性和相似性, 将语义相关性和相似性的组合作为支持向量机的输入, 构建了一个文本分类器. Zhang 等[14] 为了方便理解 HowNet 中概念之间的语义关系, 同时也为了便于计算机的处理, 在分析了 HowNet 中概念的层次关系后, 设计了一个概念 – 语义树, 并基于概念 – 语义树构建了一个词语语义相关度计算模型. 语义词典提供了规范的语义关系, 为词语语义相似度的计算带来了方便, 但是也存在如下一些问题: 1) 自然语言中的词语往往具有很强的模糊性, 一个词语往往具有很多词性、词义, 且运用场景也丰富多样, 现有的语义词典的知识表示框架很难准确、全面地表示模糊性的词语语义知识; 2) 词语语义知识含量巨大, 人工构造的语义词典相对于丰富的词语语义知识来说是很不完备的, 并且由于构造人员知识的局限性, 也很难准确地表示每个词语的客观语义事实; 3) 语义词典相对固定, 但是自然语言随着时间的变化存在一定的语义漂移现象. 这些问题都对词语语义的计算造成了一定的影响.

基于统计的方法也称为基于语料库的方法, 是建立在 "两个词语经常在同一语境中同时出现, 则这两个词语往往语义相关" 这一假设的基础之上的. 田萱等[15] 提出一种 K2CM (Keyword to concept method) 方法, 从词语 – 文档 – 概念所属程度和词语 – 概念共现程度两个方面来计算词语 – 概念的相关度. 同时文献 [15] 还指出, 基于统计的方法主要利用文档集中词语间共现性的统计数据来确定词语间的相关度, 这种方法只是利用文档中包含的内容信息, 而忽略了词语之间的具体关系以及词语相互关联的语义依据, 当统计样本不足时, 其计算结果就会出现较大误差. 近些年来, 国内外的很多研究把百科知识库 (如: 维基百科、百度百科、MBA 智库百科、互动百科等) 作为一种语料库资源融入到自然语言处理中, 取得了很好的效果. 在词语语义相关度的计算方面, Ye 等[16] 在考虑了维基百科的内容页面语义信息的基础上, 组合了维基百科的类别页面的语义信息, 提出了一种基于维基百科超链接的语义相关度计算方法. 万富强等[17] 基于中文维基百科, 将词表示为带权重的概念向量, 从而将词之间相关度的计算转化为相应的概念向量的比较, 他们在引入页面的先验概率的基础上, 利用维基百科页面之间的链接信息对概念向量的各分量值进行修正, 从而完成词语语义相关度的计算. 基于统计的方法, 把语义相关度的计算建立在大量的、可观测的语言事实上, 而不依赖于语义词典, 避免了语义词典给相关度计算带来的一些问题, 但同时也存在着对语料库依赖性大、计算量大、数据稀疏问题严重、数据噪声多、存储需求大等一些缺陷.

本文在充分研究基于语义词典的方法和基于统计的方法的优缺点的基础上, 提出了一种基于语义词典方法和语料库相结合的词语语义相关度计算模型. 首先, 在分析 HowNet 语义表示的基础上, 提取了 HowNet 中丰富的语义关系, 以语义关系三元组为存储形式, 建立基于 HowNet 的语义关系图; 然后, 在此基础上, 通过对大规模语料进行依存语法分

析, 抽取出其中存在的依存语义关系, 经过筛选后, 加入到语义关系图中, 对语义关系图做了进一步的扩展. 最后, 采用图论的相关理论对语义关系图中蕴含的语义信息进行处理, 提出一种基于语义关系图的词语语义相关度计算模型, 并通过实验验证该方法的有效性.

2　语义关系图的构建

在自然语言中, 一个词语往往具有多个含义, 在具体的语言环境中, 它们对句意的表达作用也往往是多种多样的. 同时, 词与词之间的关系更是错综复杂, 存在着各种各样的语义依赖关系, 比如同义、反义、施事、受事、句法关系等. 为了表示词语之间这些错综复杂的语义关系, 本文采用在表现复杂关系方面具有天然优势的图结构, 构建词语之间的语义关系图, 将复杂的语义关系转换为计算机可理解、可计算的数据结构. 语义关系图由结点和语义关系两部分构成, 分别对应着图中的顶点和弧. 为了构建词语间的语义关系图, 本文首先研究了 HowNet 对词语语义的表示方式, 根据 HowNet 对词语语义的表示特点, 借鉴了文献 [18] 中的方法, 提取出知网中的语义关系, 构建了基于 HowNet 的语义关系图; 然后, 通过对大规模的语料进行依存语法分析, 提取出其中的依存词语搭配, 通过相关筛选后, 将这些词语搭配及其依存关系添加到基于 HowNet 的语义关系图中, 使语义关系图得到进一步的丰富和完善.

2.1　HowNet 中的语义关系

HowNet 是一个以汉语和英语的词语所代表的概念为描述对象, 以揭示概念与概念之间以及概念所具有的属性之间的关系为基本内容的常识知识库. HowNet 采用知识系统描述语言 (Knowledge database mark-up language, KDML), 利用嵌套式的结构, 对概念以及概念的属性进行描述, 即一个复杂的概念用较简单的概念进行解释, 较简单的概念再用更简单的概念解释, 直到能够用义原表示为止. 这种结构其实质是一种隐含的树结构, 称之为概念树[18].

如 “拳台” 在 HowNet 中的描述如下:
NO. = 129348
W_C = 拳台
DEF = {facilities| 设施: domain = {boxing| 拳击},
{compete| 比赛: location = {~}},
{exercise| 锻炼: location = {~}}}

在 “拳击” 的概念描述中, KDML 表示了这样的含义: 拳台是一个设施, 这个设施所属的领域 (Domain) 是拳击领域, 这个设施是比赛的地方 (Location), 这个设施也是锻炼的地方 (Location). 也就是说, 拳台是一个用来进行拳击比赛和拳击锻炼的设施, 其所属领域是拳击领域. 将 “拳台” 这个概念用概念树重新表示如图 1 所示.

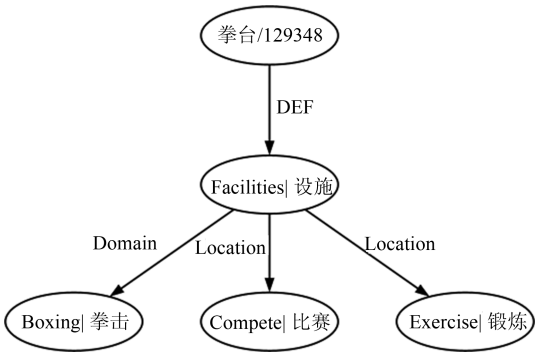


图 1　概念 “拳台” 的概念树表示

Fig. 1　The concept tree representation for “ring”

2.2　HowNet 语义关系的提取规则

通过上面的例子, 我们可以发现: 在概念树上, 每一个父结点与其子结点之间必定有一个表示语义关系的语义关系词. 因此, 在遍历概念树提取语义关系的时候, 就可以把语义关系词作为提取语义关系的标志, 即在检索语义概念树时, 当遇到语义关系词时, 考察该关系词所连接的两个结点所对应的词语是否可以与该语义关系词构成一条语义关系记录.

本文所构建的语义关系图由表示词语的结点和表示结点之间语义关系的有向边组成, 语义关系有向边由关系起始项指向关系终止项, 整个语义关系图以语义关系有向边为单位, 采用语义关系三元组 SR (关系起始项、关系终止项、语义关系词) 的方式存储, 将每一条语义关系三元组作为一条存储记录, 其存储格式如表 1 所示.

表 1　语义关系的存储格式

Table 1　The storage format of semantic relations

关系起始项	关系终止项	语义关系词
拳台	设施	DEF
…	…	…

对于一个概念描述片段 {s1 : r1 = {s2 : r2 = {s3}}}, 按照 KDML 描述规范, 每一对括号所包括的部分都是一个概念, 在该概念描述片段中三对括号所包括的内容 “{s1 : r1 = {s2 : r2 = {s3}}}”, “{s2 : r2 = {s3}}”, “{s3}” 是三个不同的概念对象.

其中 s1, s2, s3 是义原; r1, r2 是关系词, r1 是表示 s1 和 {s2:r2 = {s3}} 之间关系的关系词, r2 是表示 s2 和 {s3} 之间关系的关系词. 在提取 HowNet 中蕴含的语义关系时, 我们定义如下的规则:

规则 1. 如果关系词后面所连接的概念只是一个义原, 则直接提取语义关系. 例如: 在 {s2:r2 = {s3}}中, 若关系 r2 后的概念 "{s3}" 只是义原 "s3", 那么可以直接提取语义关系 (s2, s3, r2).

规则 2. 如果关系词后面所连接的概念是多个义原, 这时需要考察关系词后面所连接的概念是否可以用某个特定义项表示, 若可以用特定义项表示, 则可提取语义关系, 否则, 不提取该关系词的语义关系. 例如: 在 {s1:r1 = {s2:r2 = {s3}}} 中, 若 {s2:r2 = {s3}}可用义项 B 表示, 那么可提取语义关系 (s1, B, r1) 和 (s2, s3, r2); 若 {s2:r2 = {s3}}不能用某特定义项表示, 则只能提取语义关系 (s2, s3, r2).

规则 3. 如果关系词所在的整个概念可用某个义项表示时, 则可将关系词前面的义原替换为该义项并提取语义关系. 例如: 在 {s1:r1 = {s2:r2 = {s3}}} 中, {s1:r1 = {s2:r2 = {s3}}} 可用义项 A 表示, 其中的 {s2:r2 = {s3}} 可用义项 B 表示, 则可提取语义关系 (A, B, r1), (s1, B, r1), (B, s3, r2), (s2, s3, r2).

规则 4. 对于 "DEF" 关系的提取. 每个概念都需提取该概念与其第一基本义原的 DEF 语义关系. 例如: 对于图 1 所示的 "拳台" 的概念树, 需提取语义关系 (拳台、设施、DEF).

规则 5. 对于反义、对义、同义关系的提取. HowNet 中采用 Antonym Set、Converse Set、Synset Set、Taxonomy Antonym、Taxonomy Converse 5 个文件对反义、对义、同义关系进行了描述, 这三种关系直接从这 5 个文件中提取.

规则 6. 对于义原上下位关系的提取. HowNet 中的 Taxonomy entity 和 Taxonomy event 两个文件对事件和实体义原进行了描述, 其描述形式构成了树形结构, 通过对树形结构的遍历提取义原的上下位关系.

规则 7. 属性和属性值之间语义关系的提取. HowNet 中的 Taxonomy attribute value 文件对属性和属性值进行了描述, 其描述形式同样构成了树形结构, 则也通过对树形结构的遍历提取属性和属性值之间语义关系.

在研究 HowNet 收录的词语及其语义描述的过程中, 我们还发现, 其中有些词语的几个义项的中文词、词性以及概念描述等完全相同, 只有对应的英文词不同而表示为不同的义项. 由于本文所做工作的主要目的是为了计算词语的语义相关度, 与该词语的词性及其对应的英文词无关, 因此在提取语义关系之前, 我们先将 HowNet 中的中文词相同且概念描述也相同, 但编号不同的概念进行合并, 并重新为其编号, 然后再提取其中蕴含的语义关系, 构建语义关系图.

将通过 HowNet 提取出的词语之间的语义关系互相关联, 形成的网状结构称之为语义关系图 (Semantic relationship graph). 语义关系图符合图的一般特点, 具有图的一般性质, 为计算机处理语义关系提供了方便. 由于该图是以语义三元组为单位进行存储, 因此该语义关系图具有良好的可扩展性, 可以很好地融合其他语义资源中的语义关系, 进一步完善词语间的语义关联信息, 使语义关系图更加全面、客观.

2.3 语义关系图的扩展策略

在自然语言领域中, 词语以及概念由于所处的语言环境不同, 它们之间所表现出来的关系也是错综复杂的. 虽然 HowNet 着力反映了概念与概念之间以及概念所具有的属性之间的关系, 但是要想穷尽概念之间或概念所具有的属性之间的所有关系是不太可能的, 再加上人力、物力以及构造人员知识局限性的限制, HowNet 中所列举出来的关系只是最基本的、很少的一部分, 还有一些在语言使用过程中所用到的语义关系, 在 HowNet 中并没有体现出来, 或者某些词语间的语义关联方式与 HowNet 中的关联方式并不相同, 也就是在语义关系图中两个结点通过不同的路径相互连通. 为了使语义关系图中的语义关联信息更全面, 需要对基于 HowNet 构建的语义关系图做进一步的扩展, 丰富其中蕴含的语义关联信息.

在基于统计方法的相关度计算文献中[16−18] 都已指出: 如果两个词语经常同时出现在同一语境中, 则这两个词语之间往往具有一定的关联关系. 因为只有当词语间存在内在的语义关联时, 才有可能组合形成一句话并表达一个完整的句意. 另外, 依存语法认为: 句子的成分之间存在依存关系, 这种依存关系可以反映出句子中各成分之间的语义修饰关系[19]. 基于以上结论, 本文采用哈尔滨工业大学语言技术平台云上的语义依存分析接口, 对北京大学计算语言学研究所发布的《人民日报》语料进行语义依存分析, 从中提取出具有依存关系的词语搭配对, 构建词语语义关系三元组, 将这些三元组加入到基于 HowNet 语义关系图中, 实现对语义关系图的扩展. 具体的扩展策略如下:

1) 依次对人民日报语料中的每一句话进行语义依存分析, 得到每一句话的语义依存树.

2) 根据每一棵语义依存树中词语的语义依存信

息, 从中提取出实词的语义依存搭配对及其语义依存关系, 构成语义关系三元组, 并统计计算其出现的频次及其互信息[20].

3) 将频次和互信息大于一定阈值的语义关系三元组加入到基于 HowNet 的语义关系图中.

对于语义关系搭配对的共现频次和互信息的阈值选择, 采用文献 [20] 中对词语搭配选择时采用的方法, 具体的选择方法在后面的实验部分第 4.1 节进行详细讨论.

经过以上处理, 实现了对基于 HowNet 的语义关系图的扩展, 丰富了语义关系图中词语与概念的语义关联关系, 得到了相对完善的语义关系图. 在语义关系图的基础上, 就可以利用图论的相关知识和理论对词语之间错综复杂的语义关系进行处理, 实现对词语语义相关度的计算.

3　基于语义关系图的词语语义相关度计算模型

3.1　模型的基本定义

为了更好地阐述算法和便于理解算法, 下面先给出算法中将要涉及到的一些基本定义与假设.

根据图论中两点连通的概念, 本文给出语义关系图中语义连通、语义连通路径及语义连通路径长度的定义分别如定义 1、定义 2 和定义 3 所示.

定义 1 (语义连通). 在语义关系图中, 如果从结点 E_i 到 E_j 有路径存在, 则称结点 E_i 和 E_j 是语义连通的.

定义 2 (语义连通路径). 在语义关系图中, 两个语义连通的结点之间的路径称为它们的语义连通路径.

定义 3 (语义连通路径长度). 在语义关系图中, 如果结点 E_i 和 E_j 是语义连通的, 对于它们之间的某一条语义连通路径 P, 将 P 上弧的数量称为它们的语义连通路径长度, 记为 $L(E_i, E_j)$.

语义连通路径长度可用来度量结点之间语义距离, 进而确定出语义关系图中各结点所代表的词语之间语义相关度. 为此, 先引入下列的假设:

假设 1. 在语义关系图中, 如果结点 E_i 到 E_j 之间有至少一条语义连通路径, 则认为结点 E_i 与 E_j 是语义相关的.

假设 2. 在语义关系图中, 如果结点 E_i 到 E_j 不是语义连通的, 但以 E_j 为中心, 一定语义连通路径长度 α 范围内的结点构成集合 $\boldsymbol{S}$, 若集合 $\boldsymbol{S}$ 中的某个结点与 E_i 的相似度大于阈值 λ, 则认为结点 E_i 与 E_j 是语义相关的.

对于相似度阈值 λ 和语义连通路径长度 α 的选取策略将在后面的实验部分第 4.3 节进行详细讨论.

假设 1 与假设 2 共同构成了词语语义相关的必要条件, 语义关系图中的词语之间的语义相关可表示如图 2 所示. 在图 2 (a) 中, 结点 A 和 B 之间存在两条语义连通路径, 其语义连通路径长度分别为 1 和 2, 则 A 和 B 语义相关; 在图 2 (b) 中, 以结点 A 为中心, 与 A 的语义连通路径长度为 1 的结点构成集合 $\{C, E, F, H\}$, 其中 E 与 B 的相似度大于阈值 λ, 同样, 我们认为 A 和 B 是语义相关的.

3.2　模型的基本原理

本文在以上定义和假设的基础上, 为了对两个词语之间的语义相关度进行计算, 特制定以下规则.

规则 1. 在语义关系图中, 结点对自身的语义相关度为 1.

规则 2. 在语义关系图中, 如果两个语义连通的结点之间的所有语义连通路径的长度都相等, 那么, 这两个结点之间的连通路径越多, 它们的相关度越大; 反之, 相关度越小.

通过规则 2, 我们可以得出: 在连通路径长度相等的情况下, 两个词语之间的相关度大小与语义连通路径的数目成正比, 即认为相关度的值随着语义连通路径数目的增加而增大, 随着语义连通路径数目的减少而减小. 例如在图 3 (a) 中, 结点 A 和 B 的连通路径有 2 条, 且长度都为 2; 图 3 (b) 中, 结点 A 和 B 的连通路径有 3 条, 且长度也同样都为 2, 在这样情形下, 我们认为图 3 (b) 中 A 和 B 的相关度要大于图 3 (a) 中 A 和 B 的相关度, 因为在相同语义连通路径长度的前提下, 图 3 (b) 中 A 和 B 比图 3 (a) 中 A 与 B 之间存在更多的语义连通路径.

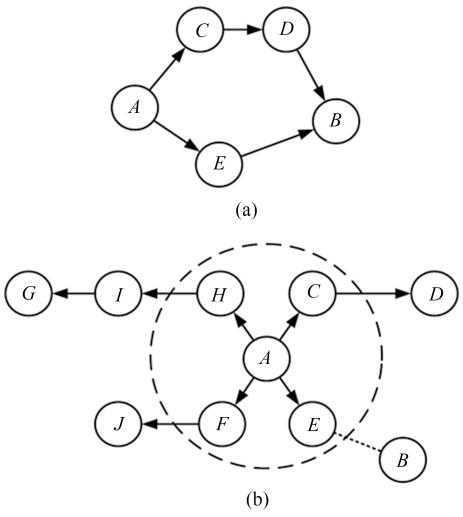


图 2　语义关系图中的语义相关

Fig. 2　The semantic relatedness in semantic relationship graph

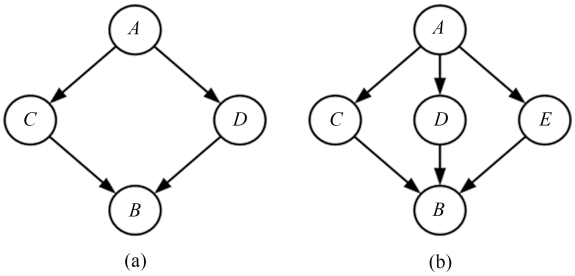


图 3 语义连通路径的数量与语义相关度的关系

Fig. 3 The relationship between the quantity of semantic connected path and semantic relatedness

规则 3. 在语义关系图中, 如果两个语义连通结点之间的连通路径数量相等, 那么, 这两个结点之间的连通路径长度越短, 它们的相关度越大; 反之, 相关度越小.

通过规则 3, 我们可以得出: 在连通路径数目相等的情况下, 两个词语之间的相关度大小与语义连通路径长度成反比, 即认为语义相关度的大小随着语义连通路径长度的增大而减小, 随着语义连通路径长度的减小而增大. 例如在图 4 (a) 中, 结点 A 和 C 的连通路径有 1 条, 且长度为 2; 图 4 (b) 中, 结点 A 和 C 的连通路径也有 1 条, 但长度为 1, 在这样的情形下, 我们认为图 4 (b) 中 A 和 C 的相关度要大于图 4 (a) 中 A 和 C 的相关度. 因为在相同数目的语义连通路径的前提下, 图 4 (b) 中 A 和 C 是直接语义连通的, 而图 4 (a) 中 A 和 C 则是依赖于其他结点语义连通的.

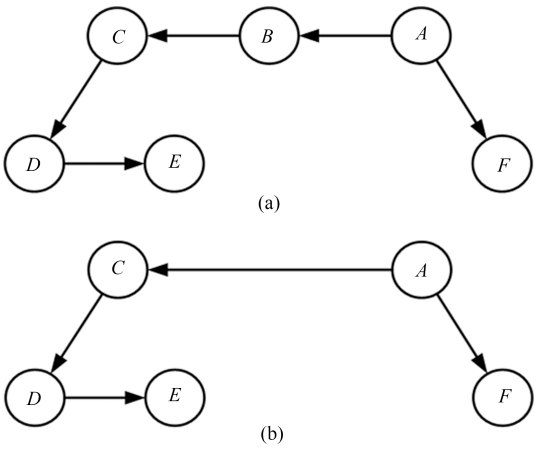


图 4 语义连通路径的长度与语义相关度的关系

Fig. 4 The relationship between the length of semantic connected path and semantic relatedness

在语义关系图中, 若两个结点之间没有语义连通路径, 则有两种可能:

1) 在构建语义关系图时, 由于语义资源的有限性, 导致语义关系图没有穷举出所有的语义关系, 致使某些语义关联缺失, 从而使得一些有关联的词语失去了语义关联, 表现在语义关系图上即为两词语的结点之间没有语义连通路径.

2) 两个词语之间本来就不是语义相关的.

对于在语义关系图中, 两个结点之间没有语义连通路径情况, 本文采用相似词替换的方法计算相关度, 具体如规则 4 所示.

规则 4. 在语义关系图中, 如果两个结点 A、B 之间没有语义连通路径, 其语义相关度计算步骤如下:

1) 以其中一个结点 A 为中心, 找出其一定长度 α 的语义连通路径内的所有结点, 构成结点集合 $\boldsymbol{S}$, 计算 $\boldsymbol{S}$ 中的每一个结点与 B 的语义相似度, 若与 B 相似度最大的结点为 C, 当 B 与 C 的语义相似度 $Sim(B,C)$ 大于阈值 λ 时, 则计算出 A 与 C 的相关度 $Rel(A,C)$; 当 $Sim(B,C)$ 小于阈值 λ 时, 记 A 与 C 的相关度 $Rel(A,C)=0$;

2) 以另一个结点 B 为中心, 采用同样的方法, 寻找结点 B 的临近结点集合中与 A 相似度最大的结点 C', 计算 A 与 C' 的语义相似度 $Sim(A,C')$ 和 B 与 C' 的语义相关度 $Rel(B,C')$;

3) 若 $Sim(B,C)$ 与 $Sim(A,C')$ 都小于阈值 λ, 则认为在以 A 或 B 为中心, 以 α 为半径的语义连通路径范围内的结点没有与 B 或 A 非常相似的词, 从而, 认为 A 与 B 不相关, 即 A 与 B 的相关度为 0; 否则, 令:

$$Rel_1(A,B)=Sim(B,C)\cdot Rel(A,C) \quad (1)$$

$$Rel_2(A,B)=Sim(A,C')\cdot Rel(B,C') \quad (2)$$

则结点 A 与 B 的语义相关度的值取 $Rel_1(A,B)$ 和 $Rel_2(A,B)$ 中的较大者, 如式 (3) 所示:

$$Rel(A,B)=\mathrm{Max}(Rel_1(A,B),Rel_2(A,B)) \quad (3)$$

其中, 词语的相似度计算方法如式 (4) 所示[21]:

$$Sim(W_1,W_2)=\max_{i=1,2,\cdots,n;j=1,2,\cdots,m}(Sim(S_{1i},S_{2j})) \quad (4)$$

式 (4) 中, 词语 W_1, W_2 分别有 n 和 m 个不同概念, S_{1i} 为 W_1 的第 i 个概念; S_{2j} 为 W_2 的第 j 个概念, $Sim(S_{1i},S_{2j})$ 表示两概念之间的相似度. 概念相似的计算方法如式 (5) 所示[21]:

$$Sim(S_1,S_2)=\sum_{i=1}^{4}\left(\beta_i\prod_{j=1}^{i}(Sim(p_{1i},p_{2j}))\right) \quad (5)$$

其中, $\beta_i\,(1\le i\le 4)$ 为可调节的参数, 分别表示 S_1 和 S_2 的第一基本义原相似度 $Sim(p_{11},p_{21})$、其他基本义原相似度 $Sim(p_{12},p_{22})$、关系义原相似度

$Sim(p_{13},p_{23})$、关系符号相似度 $Sim(p_{14},p_{24})$ 的权值系数, 且满足式 (6) 的关系:

$$\beta_1+\beta_2+\beta_3+\beta_4=1,\quad \beta_1\geq\beta_2\geq\beta_3\geq\beta_4 \tag{6}$$

$\beta_i\,(i=1,2,3,4)$ 的取值分别为[21]: 0.5, 0.2, 0.17 和 0.13. 义原的相似度计算如式 (7) 所示:

$$Sim(p_1,p_2)=\frac{2\cdot\alpha\cdot\min(dep(p_1),dep(p_2))+\lambda}{Dist(p_1,p_2)^2+2\cdot\alpha\cdot\min(dep(p_1),dep(p_2))} \tag{7}$$

其中, $dep(p_1)$, $dep(p_1)$ 分别为义原 p_1, p_2 的深度, $Dist(p_1,p_2)$ 为义原的距离, α 为可调整参数, 表示当义原相似度等于 0.5 时义原的距离, λ 同样为可调节的参数, 调节相似度整体数字的大小, α,λ 的取值为分别为 1.6 和 2.0[21].

3.3　基于语义关系图的词语语义相关度计算

本文采用图论的相关知识对语义关系图中蕴含的语义信息进行处理, 构建了基于语义关系图的词语语义相关度计算模型. 在词语语义相关度的计算过程中, 本文主要考察的是两个词语在语义关系图中的语义连通路径的数量和每条语义连通路径的长度这两个因素, 即在给定两个词语后, 通过采用图论的遍历算法, 遍历语义关系图, 得到两个词语的语义连通路径数目 n 和每条路径的长度 $L_i\,(1\leq i\leq n)$ 后, 通过 n 和 L_i 计算出两个词语的相关度.

由规则 3 可知, 当两个词语之间的语义连通路径过长时, 其语义相关度会变得很小. 在本文中, 为了强调语义连通路径长度对语义相关度计算的影响, 同时, 为了方便算法的实现, 在计算中不考虑语义连通路径长度超过 $\alpha\,(\alpha>1)$ 的语义连通路径, 并且为长度为 $1\sim\alpha$ 的语义连通路径分别赋予权值系数 $\beta_k\,(1\leq k\leq\alpha)$. 因此, 每条语义连通路径的加权长度为 $\beta_k\cdot L_i$, 其中, $k\in[1,\alpha], i\in[1,n]$. 则结点 E_i 到 E_j 之间的加权语义连通路径总长 $L(E_i,E_j)$ 如式 (8) 所示:

$$L(E_i,E_j)=\sum_{i=1}^{n}(\beta_k\cdot L_i),\quad 1\leq k\leq\alpha \tag{8}$$

同时, 考虑到语义连通路径的长度越小对语义相关度的影响力越大, 为了强调短的语义连通路径对语义相关度的影响, 将式 (8) 的加权语义连通路径总长 $L(E_i,E_j)$ 计算方式进行改进, 如式 (9) 所示:

$$L(E_i,E_j)=\sum_{i=1}^{n}\left(\left(\prod_{k=1}^{L_i}(\beta_k)\right)\cdot L_i\right),\quad 1\leq k\leq\alpha \tag{9}$$

式 (9) 中长度较小的语义连通路径对长度较大的语义连通路径起到了一定的制约作用. 其中语义连通路径长度的权值 β_k 的取值如式 (10) 所示:

$$\beta_k=\frac{k}{k+1},\quad 1\leq k\leq\alpha \tag{10}$$

由此, 可得结点 E_i 到 E_j 之间的平均加权语义连通路径长 $\overline{L(E_i,E_j)}$ 如式 (11) 所示:

$$\overline{L(E_i,E_j)}=\frac{1}{n}L(E_i,E_j) \tag{11}$$

对于词语 E_i 到 E_j, 由其在语义关系图中语义连通路径的数目和长度, 根据第 3.2 节中的相关规则, 构建词语语义相关度的计算模型, 如式 (12) 所示:

$$Rel(E_i,E_j)=\begin{cases}1, & E_i=E_j\\ \dfrac{\log_2(n+1)}{\log_2(n+1)+\overline{L(E_i,E_j)}}, & E_i\neq E_j\end{cases} \tag{12}$$

本文基于语义关系图构建了词语语义相关度的计算模型, 具体的算法过程描述如算法 1 所示:

算法1. 基于语义关系图的词语语义相关度计算算法

输入. 语义关系图 G, 语义连通长度阈值 α, 语义相似度阈值 λ, 词语 A, 词语 B

输出. 词语 A 与 B 的语义相关度 $Rel(A,B)$

过程.

步骤 1. 遍历语义关系图 G, 计算词语 A、B 的结点在 G 中的连通路径长度小于 α 的连通路径数目 n 以及每条连通路径的长度 $L_i(i\in[1,n])$, 若 $n>0$ 或者 $A=B$, 转到步骤 2; 否则, 转到步骤 3;

步骤 2. 利用式 (12) 计算 A 与 B 的相关度 $Rel(A,B)$, 转到步骤 9;

步骤 3. 以结点 A 为中心, 以长度为 α 的语义连通路径为阈值, 查找结点构成集合 $\boldsymbol{S}$;

步骤 4. 利用式 (4) 至式 (7) 计算结点 B 与集合 $\boldsymbol{S}$ 中每个结点的相似度, 得到 $\boldsymbol{S}$ 中与 B 相似度最大的结点 C;

步骤 5. 若 B 与 C 的相似度 $Sim(B,C)>\lambda$, 则利用式 (12) 计算结点 A 与 C 的相关度 $Rel(A,C)$, 否则, 记 A 与 C 的相关度 $Rel(A,C)=0$;

步骤 6. 利用式 (1) 计算 $Rel_1(A,B)$;

步骤 7. 将结点 A 和结点 B 互换, 重复以上步骤 3～步骤 6, 计算 $Rel_2(A,B)$;

步骤 8. 利用式 (3) 计算词语 A 与 B 的相关度;

步骤 9. 返回词语 A 与 B 的语义相关度 $Rel(A,B)$, 结束.

4 实验及结果分析

4.1 语义关系搭配对的阈值确定

为了确定《人民日报》语料中提取的语义关系搭配对的相关阈值, 本文参照文献 [20] 中的阈值选取策略, 同样以互信息和共现频次为阈值对语义关系搭配对进行筛选. 我们将提取的语义关系搭配对的所有搭配 (共计 452 345 个) 的互信息和共现频次提取出来, 构成了一个 $2 \times 452\,345$ 的矩阵, 将矩阵中的数据进行区间化处理, 根据它们在不同区间的分布密度, 来选择互信息和共现频次的阈值. 区间粒度的大小决定了阈值选择的精确度. 经过实验观察, 将数据区间均分为 60 等份时, 所得到的阈值对语义关系搭配对的正确性判断具有较好的区分效果. 于是我们采用 Matlab 将 $2 \times 452\,345$ 的矩阵归一化为一个 60×60 的矩阵, 矩阵中的每个值为互信息和共现频次相对应的区间范围内的词语搭配个数, 采用 Matlab 绘制 60×60 矩阵的密度分布图如图 5 所示, 密度值对语义关系搭配对的覆盖率趋势图如图 6 所示.

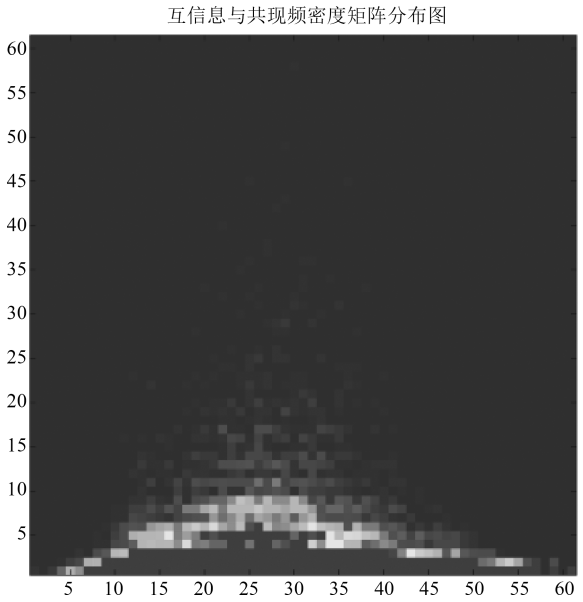


图 5 互信息与共现频次密度矩阵分布图

Fig. 5 The density matrix distribution figure between mutual information and co-occurrence frequency relatedness

通过对图 5、图 6 的分析可得, 当密度值为 905 时, 其对应密度覆盖率为 5.51 %, 通过密度矩阵分布图所选择的阈值具有较好的区分度. 我们将密度值 905 转化为互信息和共现频次的对应区间为 [0.8, 1.2] 和 [1.4, 1.9]. 基于此, 我们可以将第 2.3 节中语义关系搭配对的互信息和共现词频的阈值分别设置为 1.2 和 2. 经过随机抽取了一部分三元组, 经过人工分析发现, 采用上述方法所选择的阈值是合理的.

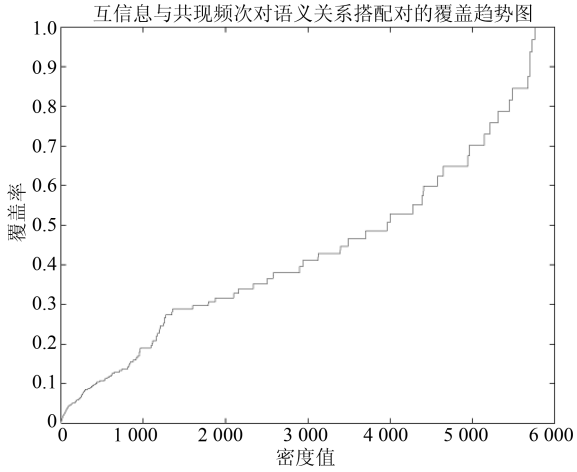


图 6 互信息与共现频次对语义关系搭配对的覆盖趋势图

Fig. 6 The coverage trend figure of mutual information and co-occurrence frequency for semantic collocation

4.2 测评数据及评价指标的构建

人工标注的数据集被认为是评价语义关系计算的 “黄金标准”, 本文的评测数据采用 Finkelstein 等[22] 构建的 WordSimilarity-353 (WS353) 数据集. WS353 数据集是英语语义计算研究中广泛应用的一个评测标准, 其中包含 353 对词语, 是当前同类公共测试集中词语量最大的数据集, 每对词语由 13 ∼ 16 个人进行手工标注, 其词语之间的语义关系以 0 ∼ 10 作为标注 (0 表示词语完全不相关, 10 表示词语密切相关), 最终的结果为人工标注的平均值. 由于 WS353 数据集为英语的词语对, 因此我们采用人工翻译的方法得到其对应的 ZWS353 中文数据集, 具体的翻译策略如下:

首先, 由两名研究生进行独立翻译, 在翻译的过程中尽量参考 HowNet 的 KDML 描述语言中的 “W_C” 字段和 “W_E” 字段间的中英文对照, 使得更多的词语能够匹配到 HowNet 中的概念. 同时在翻译的过程中, 对于一个英文词语对应于 HowNet 中的多个中文概念的, 取其中最为常见的一个概念, 对于单字词与多字词, 取多字词对应的概念. 例如 “tiger” 在 HowNet 中对应于 4 个概念, 分别是: “大虫”、“虎”、“老虎”、“戾虫”, 取其中最为常见的双字词概念 “老虎” 作为 “tiger” 的翻译.

然后, 由第三名研究生对前两名研究生独立翻译结果进行对照检查, 标记出其认为不合适的翻译.

最后, 由三名研究生共同对第三名研究生标记的不合适的翻译进行商讨, 确定最终的翻译.

本文对于最终结果的评测采用斯皮尔曼等级相关系数 (Spearman rank correlation, 简称 Spearman 系数) 进行衡量, Spearman 系数是用来估计两

个变量之间的相关性的, 其取值在 $[-1,1]$ 之间, 其值越大, 表示其相关性越大. 采用本文算法的计算结果与人工标注的结果进行对比, 求取两者的 Spearman 系数, 其值越大, 表示算法的计算结果与人工标注的结果越相似, 可认为算法的正确性越好, 同时本文也将采用 Spearman 系数与其他模型和方法进行比较, Spearman 系数的计算方法如下.

假设存在两个随机变量 $\boldsymbol{X}$、$\boldsymbol{Y}$, 它们的元素个数均为 n, 其中 X_i、Y_i 分别表示两个随机变量的第 i 个值 $(1 \leq i \leq n)$. 对 $\boldsymbol{X}$、$\boldsymbol{Y}$ 进行排序 (同时为升序或降序), 得到 $\boldsymbol{X}$、$\boldsymbol{Y}$ 的排序集合 $\boldsymbol{x}$、$\boldsymbol{y}$, 其中元素 x_i、y_i 分别为 X_i、Y_i 在 $\boldsymbol{x}$、$\boldsymbol{y}$ 中的排序序号, 令 $d_i = x_i - y_i$ $(1 \leq i \leq n)$. 则随机变量 $\boldsymbol{X}$、$\boldsymbol{Y}$ 之间的 Spearman 系数的计算如式 (13) 所示:

$$\rho = 1 - \frac{6\sum_{i=1}^{n}(d_i^2)}{n(n^2-1)} \tag{13}$$

4.3　实验结果分析

按照第 2 节所述的语义关系提取规则及语义关系图的扩展策略, 通过对 HowNet (2012) 以及《人民日报》(2000 年) 语料经过处理, 提取其中的语义关系三元组, 构造了语义关系图. 语义关系图中包括的语义关系三元组共计 836 147 条, 语义关系种类共有 168 种, 其中基于 HowNet (2012) 提取的语义关系三元组共有 524 921 条, 可以看出《人民日报》(2000 年) 语料对于语义关系图的完善起到了很大的作用.

在我们构建的词语语义相关度计算模型中, 语义连通路径长度 α 和相似度阈值 λ 都是可调节的参数. 采用本文构建的模型在 ZWS353 中文数据集进行测试, 本文模型计算出的语义相关度与人工标注的语义相关度之间的 Spearman 系数随着 α 和 λ 的变化如图 7 和图 8 所示, 根据图 7 和图 8 中 Spearman 系数的变化趋势, 我们确定当 $\alpha = 6$, $\lambda = 0.7$ 时本文提出的模型的性能最好.

同时, 从图 7 中, 可以看出, 当语义连通路径长度大于阈值后, 随着语义连通路径长度的增大, Spearman 系数会逐渐下降, 这和我们构建模型时, 固定连通路径长度的做法是高度吻合的, 也证明了第 3.2 节中的规则 3 的正确性. 从图 8 中, 可以看出, 相似度阈值 λ 取得太高 (大于 0.7) 会导致 Spearman 系数下降, 这是因为过高的相似度阈值会导致很多相关度较低的词语的相关度计算结果为 0.

为了验证本文模型的先进性, 采用 Spearman 系数对在 ZWS353 中文数据集上的测试结果进行评测, 并与现在的一些中英文词语语义相关度计算模型进行对比, 具体的结果如表 2 所示.

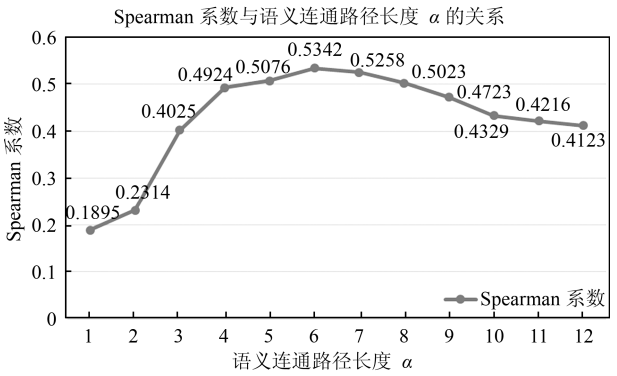


图 7　Spearman 系数与语义连通路径长度 α 关系

Fig. 7　The relationship between Spearman and semantic connected path length α

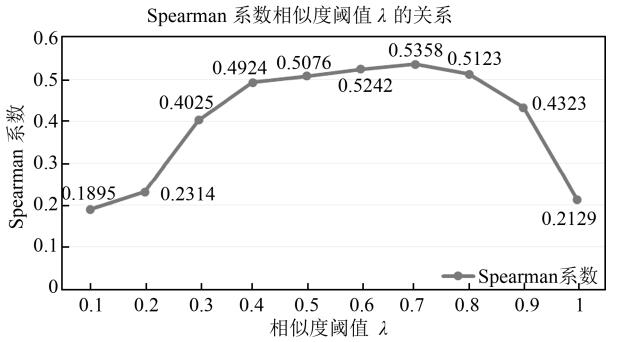


图 8　Spearman 系数与相似度阈值 λ 的关系

Fig. 8　The relationship between Spearman and similarity threshold λ

在表 2 中, 左边的数据均是在翻译的 ZWS353 中文数据集上进行的相关测评, 右边的数据则是在原始的 WS353 数据集上进行的相关测评. 其中, LIU 和 WU 都是利用 HowNet 中的义原层次体系计算词语的语义相似度, 以相似度替代相关度; TFIDF 和 COMB 都是基于维基百科的显性语义分析方法, 把词语表示为带权重的概念向量, 将词语之间的相关性计算问题转化为相应的概念向量的比较, 前者采用 TFIDF 作为词与文档的关联程度的度量, 而后者是引入了中文维基百科页面的先验概率; ICLinkBased 和 ICSubCategoryNodes 都是基于维基百科的层次分类体系来计算词语相关度, 其中 ICLinkBased 考虑的是维基百科之间的链接关系在其他文章中出现的频率, 而 ICSubCategoryNodes 考虑的是维基百科类别的子节点个数; WLM 是基于维基百科链接关系的语义相关度计算方法, 将词语映射到维基百科中的概念, 通过概念的文章之间的相关度来表示词语之间的语义相关度; WLT 是结合维基百科的链接关系与分类体系来进行词语语义相关度计算的. 对于英语的词语相关度计算,

表 2 不同方法的 Spearman 系数比较

Table 2 The comparison of Spearman in different methods

模型		Spearman 系数	模型		Spearman 系数
Knowledge-based	LIU[23]	0.4202	Knowledge-based	WUP[24]	0.3390
	WU[23]	0.3205		J&C[24]	0.3180
Corpus-based	TFIDF[17]	0.4030		Lin[24]	0.3480
	COMB[17]	**0.5150**		Resnik[24]	0.3530
	ICLinkBased[23]	0.2786	Corpus-based	LSA[24]	**0.5810**
	ICSubCategoryNodes[23]	0.2803		ESA[24]	**0.6290**
	WLM[23]	0.4984		SSA[24]	**0.5370**
	WLT[23]	**0.5126**	Knowledge + Corpus-based	WTMGW[24]	**0.7500**
Our methods	HN	0.4389			
	DSR	**0.5012**			
	HN+DSR	**0.5358**			

WUP、J&C、Lin 和 Resnik 都是从手动构造的词典 (如 WordNet) 中提取词语的相关信息进行词语的相关度计算; LSA、ESA 和 SSA 是将词语映射到维基百科中的相应文章, 采用统计的方法来计算词语的语义相关度; 而 WTMGW 是结合词典和语料库来进行词语相关度计算, 首先采用 WordNet 进行相关度的初始化, 然后采用语料库的统计信息进行迭代计算, 最终获取词语的语义相关度.

在我们的模型中, HN 表示只采用 HowNet 构建语义关系图, 进行词语的相关度计算, DSR 表示只采用大规模语料库进行依存语法分析, 构建语义关系图进行词语的相关度计算, 而 HN+DSR 是将两者结合, 进行词语的语义相关度计算.

从表 2 我们可以看出, 无论是英语还是汉语, 基于大规模语料的方法都要优于基于词典的方法, 尤其是加入维基百科语料的 COMB、WLM、WLN 模型对中文词语语义相关度的计算都有很大幅度的提高, 其 Spearman 系数基本都稳定在 0.5 左右, 其中 COMB 和 WLT 甚至超过了 0.5; 在英语中, 基于大规模语料模型的 Spearman 系数都达到了 0.5 以上. 同时, 采用词典与语料相结合的方法取得了各种模型的最好效果, 英语中 WTMGW 的 Spearman 系数达到了最高的 0.75, 本文提出的模型也达到了 0.5358, 为中文模型中的最优.

同时, 在我们的模型中, HN 模型与 DSR 模型的性能低于 HN+DSR 模型, 并且 HN 模型的性能低于 DSR 模型, 这与上面分析出的基于大规模语料的方法优于基于词典的方法且词典与语料相结合的方法效果最好的结论是吻合的. 在我们的模型中, 采用的词典为 HowNet, 由于 HowNet 是一个常识知识库, 因此从 HowNet 中提取出的语义关系覆盖面比较广, 但与实际的语言使用情况有一定的差异. 而对于《人民日报》采用依存语法分析, 提取出的语义关系比较贴近于真实的语言使用环境, 但具有一定的领域性. 在 HN+DSR 模型中, 对于一些在实际语言环境中经常使用的相关词语搭配, 其相关度的计算主要来自于对大规模语料进行依存语法的分析得到的语义关系, 例如: “新年” 和 “音乐会” 两个词语在通过 HowNet 构建的语义关系图中并不存在语义连通路径, 但通过对语料库的依存语法分析发现两者是存在语义关系的: (音乐会、新年、Nmod) (其中, Nmod 表示名字修饰角色的语义关系); 但是, 对于一些反义、对义、同义、上下位关系及属性和属性值之间的关系, 由于 HowNet 中有专门的描述文件, 对于这方面的词语语义相关度计算起到了不少的作用.

另外, 我们也可以看出, 虽然是类似的模型, 但中文模型的性能要略差于英文模型. 本文的模型与英文中的 WTMGW 模型的性能也有很大的差距, 甚至与英文中基于语料库的模型也有一些差距, 其主要原因可能是在对 WS353 数据集的翻译过程中引入了误差. 因为翻译的过程中, 很多英文单词在翻译为中文时, 对应着很多的中文翻译, 而且各个翻译之间的差距很大, 很难取舍, 例如: 单词 “stock”, 对应到 HowNet 中的概念有 “库存”、“储备”、“供应”、“股票”、“股份”、“原汤”、“砧木”, 这些给我们的翻译造成了一定的阻碍, 也给我们实验的性能造成了干扰.

为了进一步验证本文模型的可用性, 我们从构建的语义关系图中抽取了 10 个实词, 每两个组成一组测试数据, 构建了一个包含 100 组词语对的实际测试数据集, 采用 HN+DSR 模型进行测试, 其部分

实验结果如表 3 所示.

表 3　语义相关度计算的实验结果

Table 3　The experimental result of semantic relatedness computation

词语 1	词语 2	相关度
足球比赛	比分	0.9004
足球比赛	直播	0.8438
足球比赛	场地	0.6034
足球比赛	规则	0.7925
足球比赛	法庭	0.2016
滑冰	足球比赛	0.2415
滑冰	流畅	0.7924
滑冰	速度	0.8415
滑冰	摔倒	0.7524
滑冰	法庭	0.2965
足球比赛	流畅	0.0251

由表 3 中数据可以看出, 绝大部分结果还是比较符合习惯上对相关度的主观判断的, 且实验结果比较平稳, 不会出现极端值的问题. 但从实验结果也可以看出, 部分结果还不够理想, 例如: “滑冰” 和 “法庭” 的相关度比 “足球比赛” 和 “法庭” 的相关度稍高. 导致部分相关度不太准确的原因主要有以下几点:

1) HowNet 中有些词语的义原描述不够合理, 导致词语间的语义关系产生了误差. 如 “比分” 的第一义原为 “符号”, 这将会导致 “比分” 和 “符号” 两个词的相关度计算结果的偏差.

2) 在通过语义依存分析器分析《人民日报》语料, 可能会分析出一些错误的语义依存搭配关系, 同时, 还有一些词语在某些特定的语义情况下存在语义依存关系, 但其本身的语义相关度并不大, 例如: 在《人民日报》语料中存在大量类似 “新华社北京十二月三十一日电” 的语句, 在这样的语义环境中, “新华社” 和 “电” 存在 Orig (源事关系) 语义关系, 但其两者之间的语义相关性并不强烈.

3) 虽然本文的模型综合使用了语义词典和大规模的语料库, 有效地避免了两种模型单独使用时的某些弊端, 但是依然存在数据资源有限、数据稀疏、词义漂移等问题, 这些为词语语义相关度的计算造成了干扰.

5　结论及工作展望

本文在分析现有的词语语义相关度计算模型的基础上, 提出了一种语义词典和语料库资源相结合的词语语义相关度计算模型. 首先, 以 HowNet 中概念与概念之间以及概念所具有的属性之间的语义关系和大规模语料中统计出的词语语义依存关系为基础, 构建了一张语义关系图, 然后, 利用图论的相关算法和理论对语义关系图中的语义依存关系进行处理, 提出了一种基于语义关系图的词语语义相关度计算模型. 实验表明, 本文模型计算得到的词语语义相关度结果较为合理.

在接下来的工作中, 我们计划增大语料库的数据量, 进一步丰富语义关系图中的语义关联信息, 探索更为直接的语义三元组获取方法, 避免由于语义词典和语义依存分析的错误传递而导致词语语义相关度计算的偏差, 同时更进一步地完善词语语义相似度的计算模型, 期望得到更加真实有效的词语语义相关度.

References

1 Gracia J, Mena E. Web-based measure of semantic relatedness. In: Proceedings of the 9th International Conference on Web Information Systems Engineering. Auckland, New Zealand: Springer, 2008. 136−150

2 Resnik P. Using information content to evaluate semantic similarity in a taxonomy. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence. Montreal, Quebec, Canada: Morgan Kaufmann Publishers Inc., 1995. 448−453

3 Liu H W, Xu J J, Zheng K, Liu C F, Du L, Wu X. Semantic-aware query processing for activity trajectories. In: Proceedings of the 10th ACM International Conference on Web Search and Data Mining. Cambridge, UK: ACM, 2017. 283−292

4 Ensan F, Bagheri E. Document retrieval model through semantic linking. In: Proceedings of the 10th ACM International Conference on Web Search and Data Mining. Cambridge, UK: ACM, 2017. 181−190

5 Liu Kang, Zhang Yuan-Zhe, Ji Guo-Liang, Lai Si-Wei, Zhao Jun. Representation learning for question answering over knowledge base: an overview. *Acta Automatica Sinica*, 2016, **42**(6): 807−818
(刘康, 张元哲, 纪国良, 来斯惟, 赵军. 基于表示学习的知识库问答研究进展与展望. 自动化学报, 2016, **42**(6): 807−818)

6 Zhang Y M, Iwaihara M. Evaluating semantic relatedness through categorical and contextual information for entity disambiguation. In: Proceedings of the IEEE/ACIS 15th International Conference on Computer and Information Science. Okayama, Japan: IEEE, 2016. 1−6

7 Li C, Bendersky M, Garg V, Ravi S. Related event discovery. In: Proceedings of the 10th ACM International Conference on Web Search and Data Mining. Cambridge, UK: ACM, 2017. 355−364

8 Arab M, Jahromi M Z, Fakhrahmad S M. A graph-based approach to word sense disambiguation. An unsupervised method based on semantic relatedness. In: Proceedings of the 24th Iranian Conference on Electrical Engineering. Shiraz, Iran: IEEE, 2016. 250−255

9 Xin Yu, Xie Zhi-Qiang, Yang Jing. Semantic community detection research based on topic probability models. *Acta*

Automatica Sinica, 2015, **41**(10): 1693−1710
(辛宇, 谢志强, 杨静. 基于话题概率模型的语义社区发现方法研究. 自动化学报, 2015, **41**(10): 1693−1710)

10 Budanitsky A, Hirst G. Evaluating WordNet-based measures of lexical semantic relatedness. *Computational Linguistics*, 2006, **32**(1): 13−47

11 Taieb M A, Aouicha M B, Hamadou A B. A new semantic relatedness measurement using WordNet features. *Knowledge and Information Systems*, 2014, **41**(2): 467−497

12 Liu Qun, Li Su-Jian. Word similarity computing based on HowNet. *Computational Linguistics*, 2002, **7**(2): 59-76
(刘群, 李素建. 基于《知网》的词汇语义相似度计算. 中文计算语言学, 2002, **7**(2): 59−76)

13 Zhang P Y. A HowNet-based semantic relatedness kernel for text classification. *TELKOMNIKA*, 2013, **11**(4): 1909−1915

14 Zhang G P, Yu C, Cai D F, Song Y, Sun J G. Research on concept-sememe tree and semantic relevance computation. In: Proceedings of the 20th Pacific Asia Conference on Language, Information and Computation. Wuhan, China: Tsinghua University Press, 2006. 398−402

15 Tian Xuan, Du Xiao-Yong, Li Hai-Hua. Computing term-concept association in semantic-based query expansion. *Journal of Software*, 2008, **19**(8): 2043−2053
(田萱, 杜小勇, 李海华. 语义查询扩展中词语−概念相关度的计算. 软件学报, 2008, **19**(8): 2043−2053)

16 Ye F Y, Zhang F, Luo X F, Xu L Y. Research on measuring semantic correlation based on the Wikipedia hyperlink network. In: Proceedings of the IEEE/ACIS 12th International Conference on Computer and Information Science. Niigata, Japan: IEEE, 2013. 309−314

17 Wan Fu-Qiang, Wu Yun-Fang. Computing lexical semantic relatedness with Chinese Wikipedia. *Journal of Chinese Information Processing*, 2013, **27**(6): 31−38
(万富强, 吴云芳. 基于中文维基百科的词语语义相关度计算. 中文信息学报, 2013, **27**(6): 31−38)

18 Wang Hong-Xian, Zhou Qiang, Wu Xiao-Jun. The automatic construction of lexical semantic relationship graph based on HowNet. *Journal of Chinese Information Processing*, 2008, 22(5): 90−96
(王宏显, 周强, 邬晓钧. 《知网》语义关系图的自动构建. 中文信息学报, 2008, 22(5): 90−96)

19 Zheng Li-Juan, Shao Yan-Qiu, Yang Er-Hong. Analysis of the non-projective phenomenon in Chinese semantic dependency graph. *Journal of Chinese Information Processing*, 2014, **28**(6): 41−47
(郑丽娟, 邵艳秋, 杨尔弘. 中文非投射语义依存现象分析研究. 中文信息学报, 2014, **28**(6): 41−47)

20 Zhang Yang-Sen, Zheng Jia. Study of semantic error detecting method for Chinese text. *Chinese Journal of Computers*, 2016, **39**, Online Publishing No. 122
(张仰森, 郑佳. 中文文本语义错误侦测方法研究. 计算机学报, 2016, **39**, 在线出版号 No. 122)

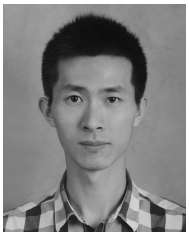
21 Zhang Hu-Yin, Liu Dao-Bo, Wen Chun-Yan. Research on improved algorithm of word semantic similarity based on HowNet. *Computer Engineering*, 2015, **41**(2): 151−156
(张沪寅, 刘道波, 温春艳. 基于《知网》的词语语义相似度改进算法研究. 计算机工程, 2015, **41**(2): 151−156)

22 Finkelstein L, Gabrilovich E, Matias Y, Rivlin E, Solan Z, Wolfman G, Ruppin E. Placing search in context: the concept revisited. *ACM Transactions on Information Systems*, 2002, **20**(1): 116−131

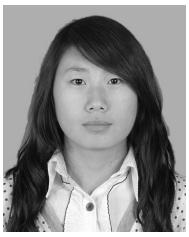
23 Wang Xiang, Jia Yan, Zhou Bin, Ding Zhao-Yun, Liang Zheng. Computing semantic relatedness using Chinese Wikipedia links and taxonomy. *Journal of Chinese Computer Systems*, 2011, **32**(11): 2237−2242
(汪祥, 贾焰, 周斌, 丁兆云, 梁政. 基于中文维基百科链接结构与分类体系的语义相关度计算. 小型微型计算机系统, 2011, **32**(11): 2237−2242)

24 Liu B Q, Feng J, Liu M, Liu F, Wang X L, Li P. Computing semantic relatedness using a word-text mutual guidance model. In: Proceedings of the 3rd CCF Conference on Natural Language Processing and Chinese Computing. Shenzhen, China: Springer, 2014. 67−78

张仰森 北京信息科技大学教授. 主要研究方向为自然语言处理和人工智能. 本文通信作者.
E-mail: zhangyangsen@163.com
(**ZHANG Yang-Sen** Professor at the Beijing Information Science and Technology University. His research interest covers nature language processing and artificial intelligence. Corresponding author of this paper.)



郑 佳 北京信息科技大学硕士研究生. 主要研究方向为自然语言处理.
E-mail: zhengjia0826@163.com
(**ZHENG Jia** Master student at the Beijing Information Science and Technology University. His main research interest is nature language processing.)



李佳媛 北京信息科技大学硕士研究生. 主要研究方向为自然语言处理.
E-mail: ljyuan0616@126.com
(**LI Jia-Yuan** Master student at the Beijing Information Science and Technology University. Her main research interest is nature language processing.)