情报科学
第37卷 第3期 2019年3月

文本相似度计算方法研究综述

王春柳,杨永辉,邓　霏,赖辉源

(中国工程物理研究院 计算机应用研究所,四川 绵阳 621000)

摘　要:【目的/意义】文本相似度计算是自然语言处理中的一项基础性研究,通过总结和分析文本相似度计算的经典方法和当前最新的研究成果,完善对文本相似度计算方法的系统化研究,以便于快速学习和掌握文本相似度计算方法。【方法/内容】对过去20年的文本相似度计算领域的经典文献进行整理,分析不同计算方法的基本思想、优缺点,总结每种计算方法的侧重点和不同方向上最新的研究进展。【结果/结论】从表面文本相似度计算方法和语义相似度计算方法两方面进行阐述,形成较为全面的分类体系,其中语义相似度计算方法中的基于语料库的方法是该领域最为主要的研究方向。

关键字: 文本相似度;语义相似度;语料库

中图分类号:G254; G252.8　　**DOI:**10.13833/j.issn.1007-7634.2019.03.026

A Review of Text Similarity Approaches

WANG Chun-liu, YANG Yong-hui, DENG Fei, LAI Hui-yuan

(Institute of Computer Application, China Academy of Engineering Physics, Mianyang 621000,*China)*

Abstract:【Purpose/significance】Text similarity calculation is a basic research in natural language processing. Through summing up and analyzing the classical methods of text similarity calculation and the latest research results, we improve the systematic research on text similarity algorithms, so as to quickly learn and grasp the text similarity calculation methods. 【Method/process】We collate the classical literature in the field of text similarity algorithms in the past 20 years, and analyze the basic ideas, advantages and disadvantages of different computing methods, and summarizes the emphasis of each method and the latest research progress in different directions.【Result/conclusion】The surface text similarity calculation method and semantic similarity calculation method were discussed to form a more comprehensive classification system. Corpus-based approach to semantic similarity calculation is the most important research direction in this field.

Keywords: text similarity; semantic similarity; sorpus-based; review

1 引　言

文本相似度计算是指通过一定的策略比较两个或多个实体(包括词语、短文本、文档)之间的相似程度,得到一个具体量化的相似度数值。随着计算机技术的迅速发展,越来越多的信息充斥在网络平台上,对这些文本信息的深度挖掘和研究对于帮助人们快速准确获取与需求相关的内容具有非常实际的意义。其中,文本相似度算法是文本挖掘中的一个至关重要的算法,是联系文本建模和表示等基础研究和文本潜在信息上层应用研究的纽带[1]。例如,在文本分类、文本聚类、词义消歧等信息检索问题上,搜索引擎中的问答系统、智能检索等问题都需要文本相似度算法作为支撑。此外,文本相似度算法也广泛应用在自动摘要、机器翻译等自然语言处理问题中,是自然语言处理问题中的核心算法。因此,完善对文本相似度算法的系统化研究具有非常重要的应用价值。

目前,文本相似度计算的方法已经越来越多,大多数学者比较认可的分类方式是:基于字符串(String-Based)的方法、基于语料库(Corpus-Based)的方法、基于知识库(Knowledge-Based)的方法和混合方法[2-4]。为便于快速学习文本相似度计算方法,本文将近年来的国内外相关文献进行收集、

收稿日期:2018-05-27

基金项目:国防基础科研计划重点项目(JCKY2016212B004)

作者简介:王春柳(1993-),女,吉林辽源人,硕士研究生,主要从事语义计算、对话系统评测研究.

整理和分析，在不影响全局分类的情况下，对各类方法重新进行了归纳、梳理和补充，从表面文本相似度(Surface Text Similarity)计算方法、语义相似度(Semantic Similarity)计算方法两方面进行阐述，使得分类结构更浅显易懂。由于混合方法是将文本相似度计算各类中的不同方法进行结合计算来提高计算效果，本文将不再赘述关于混合方法的研究内容。

2 表面文本相似度计算

表面文本相似度计算直接针对原始文本，作用于字符串序列或字符组合，以两个文本的字符匹配程度或距离作为相似度的衡量标准。其算法原理简单、易于实现，是研究历史最长的一类文本相似度算法。下面我们主要针对Gomaa[4]和陈二静[5]等人的分类内容进行补充和完善，其整体的分类体系如图1所示。

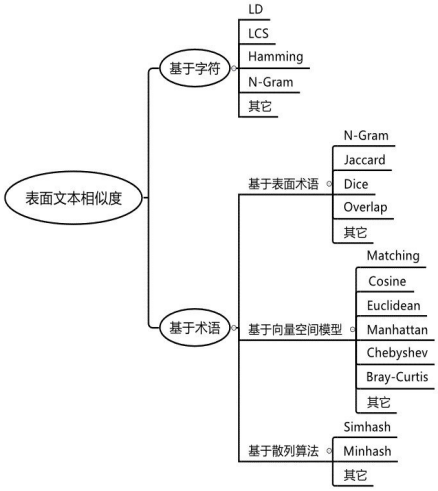

图1 表面文本相似度计算方法分类

根据计算粒度的区别，通常将表面文本相似度计算方法细分为基于字符(Character-Based)的方法和基于术语(Term-Based)的方法。基于字符的方法包括编辑距离(Levenshtein Distance，LD)[6]、最长公共子序列(Longest Common Sequence，LCS)[7]、汉明距离(Hamming Distance)[8]、N元模型(N-Gram)[9]等，其中关于编辑距离的算法还存在很多种变体，如Weighted- Levenshtein、Damerau- Levenshtein[10]、Optimal String Alignment、Jaro-Winkler[11]等，在拼音纠错和链接领域都有着广泛的应用，而Needleman-Wunsch[12]算法和Smith-Waterman[13]算法属于LSC中的一类，是基于动态规划的思想对两个序列分别进行全局最优比对和局部最优比对，主要应用在生物信息学中的DNA序列比对。

基于术语的方法根据计算时表示方式的不同可以分为基于表面术语(Surface Term/word)和基于空间向量模型(Vector Space Model，VSM)[14]以及基于散列(Hash)算法三种方式，其中N-Gram、Jaccard相似性[15]、Dice系数[16]、重叠系数(Overlap Coefficient)是直接计算术语的匹配程度，其核心思想是将文本相似性问题转化为集合的问题。

基于向量空间模型的方法包括匹配系数(Matching Coefficient)、余弦相似度(Cosine)、欧式距离(Euclidean Distance)、曼哈顿距离(Manhattan Distance)[17]、切比雪夫距离(Chebyshev Distance)、布雷柯蒂斯相异性(Bray-Curtis Dissimilarity)[18]等，其中曼哈顿距离、欧式距离和切比雪夫距离可以统一表示为明可夫斯基距离(Minkowski Distance)。这种方法是将术语表示成向量后再进行计算，这里的向量是指通过词频-逆文档频率(Term Frequency-Inverse Document Frequency，TF-IDF)模型将两个文本分别表示为 $\vec{x}$ 和 $\vec{y}$ 的矢量形式[19]，或者直接通过最简单的词集模型(Set Of Words，SOW)将文本表示为独热向量(One-hot vector)形式，这里的向量都不具有语义信息，仅是简单地将文本表示为向量形式以便于运算。除此之外，通过将文本向量表示看作是不同的变量然后使用统计相关系数Pearson、Spearman和Kendall也可以计算文本相似性[5]。下面我们通过表格的方式将上述一些重要的方法及定义列入表1中。

基于散列算法的文本相似度方法包括局部敏感哈希(Locality Sensitive Hashing，LSH)算法和局部保留哈希(Locality Preserving Hashing，LPH)算法，这两类算法主要针对最近邻搜索问题。其中Simhash和Minhash是两个广泛应用于大规模数据处理的局部敏感散列算法[20]。传统的Hash算法是将原始文本随机映射成唯一的签名值，根据签名值只能判断

表1 表面文本相似度计算方法

方法	定义
LD	两个字符串之间由一个转变为另一个所需的最少编辑操作次数，包括插入、删除和替换。
LCS	两个字符串的最长公共连续子序列，可以计算其与较短或较长的字符串的长度的比值(LCSR)。
Hamming	两个等长字符串在对应位置上不同的数量，主要用于通信编码领域。
N-Gram	两个文本的相同N元组数量与总N元组数量的比值，对象可以是字符的N元组或术语的N元组。
Jaccard	$S_{jacc}=\lvert A\cap B\rvert/\lvert A\cup B\rvert$，集合思想，两个文本中相同词语的个数与全部非重复词语的个数的比值。
Dice	$S_{dice}=2*\lvert A\cap B\rvert/(\lvert A\rvert+\lvert B\rvert)$，集合思想，两个文本中相同词语个数的二倍与各个文本非重复词语个数之和的比值。
Overlap	$S_{over}=\lvert A\cap B\rvert/\min(\lvert A\rvert,\lvert B\rvert)$，集合思想，如果两个文本一个是另一个的子集，则认为两个文本是完全相似的 。
Matching	非常简单的基于向量的方法，只计算两个向量相同项都是非零的个数。
Cosine	$S_{\cos}=\vec{x}\cdot\vec{y}/(\lVert\vec{x}\rVert\cdot\lVert\vec{y}\rVert)$，计算两个向量的夹角的余弦值。
Manhattan	$M(\vec{x},\vec{y})=\sum_{i=1}^{n}\lvert x_i-y_i\rvert$，两个向量对应坐标值的差异之和。
Euclidean	$E(\vec{x},\vec{y})=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}$，两个向量对应坐标值之间的平方差之和的平方根。
Chebyshev	$C(\vec{x},\vec{y})=\max_i(\lvert x_i-y_i\rvert)$，两个向量对应坐标值差的绝对值的最大值。
Bray-Curtis	$B(\vec{x},\vec{y})=\sum_{i=1}^{n}\lvert x_i-y_i\rvert/\sum_{i=1}^{n}\lvert x_i+y_i\rvert$，常用于生物信息学中表征两个群落的差异性。

出两个文本是否相同而无法判断两个文本是否相似[21]。由于上述基于集合和基于向量空间模型的两类算法针对大规模数据普遍存在开销大、效率低和准确率差的问题，所以针对海量数据去重问题Simhash应运而生，该算法由Charikar等人于2002年提出[21]，被认为是目前最好、最有效的网页相似内容去重算法[22]。Google在2006年进行了一次大规模的评估，比较了Minhash算法和Simhash算法的性能[23]。2007年，Google公开了关于使用Simhash对网页爬行进行重复检测[24]，并使用Minhash和LSH实现新闻个性化的成果[25]。2014年，Shrivastava等人通过实验证明了在高相似度区域Minhash明显优于Simhash[20]。

值得一提的是在英文中每个单词即为一个术语，在使用上述方法计算之前通常需要进行词形归一、去除停止词、大小写转换等预处理操作，中文处理相对英文较为复杂，需要进行分词，分词结果的准确性对于计算结果的好坏会有很大的影响。此外，许多学者如Zhao[19]、JCS[3]等人将文本的长度信息加入到相似性计算中，提出了基于字长的相似度计算方法(Text Difference Measures)，将字长信息作为相似度计算结果的一个重要影响因子，并通过实验证明了该方法的有效性。

表面文本相似度计算方法是相对原始并直观的计算方法，实现非常简单且不需要其他资源的支撑，是目前许多算法的计算基础，如对话系统评测、机器翻译和自动摘要评估中用到的ROUGE[26]、BLEU[27]等评测算法都是基于上述基本方法。由于这种方法是直接对原始文本进行粗略的匹配或计算距离，并未考虑词语本身的含义和词语之间的关系，所以它不适用于词语之间的相似度计算，也无法很好的计算带有多义词(同一个词可以有多个含义)和同义词(两个词可以代表同一个概念)的文本。

目前比较普遍的一个做法是将上述部分方法计算的结果作为文本相似度的特征，使用机器学习算法对这些特征进行学习来得到最终的相似度计算结果。例如，Zhao[19]等人在2014年的实验中使用了监督学习和半监督学习算法，利用上述的部分表面文本相似度方法所得的结果作为机器学习算法的输入，通过对几种结果较优的机器学习算法的结果计算平均值来获取文本最终的相似度结果。Eyecioglu等人在2015年的SemEval比赛中仅使用了基于字重叠和字符重叠的N-Gram结果作为特征，并采用一个SVM分类器进行训练，最终得到了与使用更复杂的NLP处理工具和外部资源的方法相媲美的结果[28]。

3 语义相似度计算

上述的表面文本相似度计算方法只考虑表层字词而不考虑这些字词在句子中真实的含义，针对这一问题，研究人员又进一步提出了语义相似性计算的方法。语义相似度的计算主要包括两大类：基于知识库(Knowledge-Based)与基于语料库(Corpus-Based)两种[3]。

3.1 基于知识库

基于知识库的相似度计算方法是通过使用知识库中获取的信息来量化两个文本在语义上的关联程度[29]，主要是基于概念间结构层次关系组织的语义词典的方法，根据在这类语言学资源中概念之间的上下位和同位关系来计算词语的相似度[30]。基于知识库的分类可以按知识库的类型分为基于本体(Ontology-Based)和基于网络知识两种[7]，也有学者根据知识库的结构差异表述为基于树状结构的相似度计算和基于有向图结构的相似度计算[31]。其整体的分类体系如图2所示。

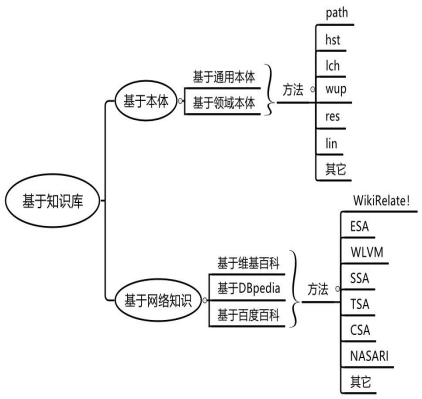

图2 基于知识库的语义相似度计算方法分类

3.1.1 基于本体

基于本体的相似度计算方法主要是基于语义词典进行计算。目前，国内主要使用的语义词典为HowNet(《知网》)，中国的许多学者在这方面都取得了一定的进展，目前仍不失为一个研究热点。还有一部分研究使用的语义词典是《同义词词林》，《同义词词林》是目前国内上在结构上与国际上著名的英文语义词典WordNet最为接近的词典，但是国内提出的基于《同义词词林》的相似度计算方法的准确度都比较低，与国际上优秀的基于路径距离的算法相比还有很大的距离[29]。上述提到的语义词典均属于通用本体，通用本体很少包含专有名词、新词、俚语和特定领域的技术词汇等，而针对一些特定任务，通用本体往往不能满足我们的需求，所以基于领域本体的计算方法在当前的研究工作中也占有一席之地，领域本体的种类也非常多，包括地理本体、体育本体[32]、医学本体[33]等，在近年来的研究中都曾被使用到。

目前，国际上主流的分类方式是将基于本体的语义相似度计算分为基于本体结构(Ontology Structure-Based)、基于信息内容(Information content-based)、基于属性(Feature-based)以及混合式的语义相似度计算，这四类算法都是建立在“IS_A”关系树状分类体系基础上，如WordNet语义词典，而中文的语义词典因为与WordNet结构不尽相同，所以计算方法也有所不同。基于信息内容的方法和基于本体结构(Ontology Structure-Based)下的基于路径距离(Edge-counting Measures/Path Length Based)的方法是目前研究最为广泛的两部分，其中Shortest Path(path)、Leacock &Chodorow(lch)、

Wu&Palmer(wup)、Hirst&St-Onge(hst)方法和Resnik(res)、Lin(lin)、Jiang&Conrath(jcn)方法分别是基于路径距离和基于信息内容中的典型算法[4]。在Poorna等人2018年的研究成果中,选取了基于路径距离和基于信息内容中的六种方法进行对比实验,证明wup、hst和lin三种方法的计算准确度优于其他方法[32],类似的研究还有Althobaiti等人在医学文本数据上对上述方法的实现和对比[33]。需要注意的是,这些算法在不同领域本体的计算结果优劣各不相同,所以实验中关于算法优劣的结论并不具有普遍性。

基于本体结构的方法是通过两个概念词在本体树状分类体系中的路径长度和树的节点深度来量化它们之间的语义距离,该算法计算复杂度相比于其他几种算法最小,但需假设本体分类体系中所有的边都同等重要,而这一假设并不成立。基于信息内容的方法是通过被比较概念词的公共父节点概念词所包含的信息内容来衡量它们之间的相似度,能综合反映概念在句法、语义、语用等方面的相似性和差异,但这类算法比较依赖于训练所用的语料库,如果针对领域本体的语料库不全或数据噪声比较大,那么算法的实施和计算准确度都会有影响。基于属性的语义相似度计算的基本思想是事物由其属性特征反映其本身,事物之间的关联程度与它们所共有的公共属性数相关,此方法必须依赖于概念具备完备的属性集。而混合式语义相似度实质上是对上述三种算法的综合考虑[34],虽然能够一定程度上提高计算结果的准确性,但是并没有从根本上克服它所基于的方法的局限性。

对于处理中文文本相似度问题时所使用的基于HowNet的词语语义相似度计算的基本方法主要可以概括为三个步骤:义原相似度计算、概念相似度计算和词语相似度计算。义原相似度计算主要是利用HowNet中的词语的义原层次的语义距离来计算相似度,其中义原是词典中最基本、最不易分割的意义的最小单位。概念相似度计算也称为义项相似度计算,一个词语可以有多个义项,而一个义项是由多个义原组成,所以义项间的相似度计算转化为了义原间的相似度计算。词语的相似度计算是取其义项所有组合中相似度的最大值[34]。进一步的,基于HowNet的句子相似度的计算则是以词语语义相似度计算为基础,具体的计算方式如下[35]:

设句子A和B分词和预处理后的词序列分别为$A(A_1,A_2,\cdots,A_m)$和$B(B_1,B_2,\cdots,B_n)$,定义句子中任意两个词$A_i(1\leq i\leq m)$和$B_j(1\leq j\leq n)$的相似度为$Sim(A_i,B_j)$。句子A和B之间的语义相似度$Sim(A,B)$为:

$$Sim(A,B)=(\frac{\sum_{i=1}^{m}a_i}{m}+\frac{\sum_{j=1}^{n}b_j}{n})/2 \quad (1)$$

其中,

$$a_i=\max(Sim(A_i,B_1),Sim(A_i,B_2),\cdots,Sim(A_i,B_n)) \quad (2)$$

$$b_j=\max(Sim(A_1,B_j),Sim(A_2,B_j),\cdots,Sim(A_m,B_j)) \quad (3)$$

与其他相似度算法不同,基于本体的相似性度量需要许多NLP资源如词性标注、词法数据库,词语列表等,因此许多语言由于资源不足而仍处于发展阶段[3]。该方法的原理相比于基于语料库的计算方法更直观、更易于理解,但它非常依赖于预先建立好的语义词典,无法适应层出不穷的新词汇,受限于概念层次网络,网络的层次结构将直接影响到相似度计算的结果。另外,这种方法虽然能比较准确地反映词语之间语义方面的相似性和差异,但对于整个句子或文档中的词语顺序和句法都考虑地比较少。再者,本体由专家参与建设,对于其内容的更新和维护都非常耗时耗力,而且其内容受人们的主观影响使其不能很好的反映客观现实。

3.1.2 基于网络知识

维基百科作为全世界最大的多语种、开放式的在线百科全书,相比于语义词典,其覆盖范围更加广泛,知识描述更加全面,信息内容更新更加迅速,因此目前基于知识库的研究中有一部分是基于维基百科进行的。维基百科具有较好的结构化信息,被称为是半结构化的知识库。可以将维基百科看作是两个巨大的网络,一是由页面组成的网络(页面网),每个节点表示一个网页,每条边表示一个链接,不同的网页之间通过出链和入链连接在一起,二是由类别组成的网络(类别网),每一个矩阵框代表维基百科的一个类,不同的类别通过子类和父类相互连接在一起,如图3所示。这两个网络都可以抽象成有向图,对维基百科页面网和类别网的处理也就抽象成了对有向图的处理[30]。

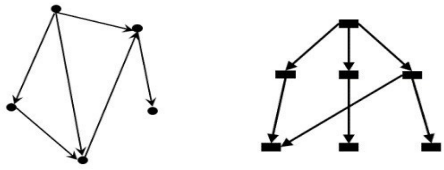

图3 维基百科的网页面(1)和类别面(2)示例图

基于维基百科的代表算法包括WikiRelate![36]、显示语义分析(Explicit Semantic Analysis,ESA)[37]和Wikipedia Link-based Measure(WLM)[38]。WikiRelate!方法将维基百科的文档类型结构代替语义词典的概念层次结构,将维基百科的文档内容代替语义词典的词汇定义,模仿基于语义词典的度量方法进行计算,该方法仅限于词对之间的相似度计算。ESA方法可以比较任意长度的文本相似度,它模仿信息检索中使用的向量空间模型,利用维基百科中的链接信息将文本映射为TF-IDF的加权向量形式,通过向量之间的余弦相似度来获取文本之间的相似度。WLM也是基于向量空间模型的方法,但它只计算维基百科文档之间的链接信息,而不考虑文本内容,这种计算方法非常简单,但同时准确率也不如ESA方法。

在上述方法的基础上,研究学者又陆续提出了WikiWalk[39]方法、显著语义分析(Salient Semantic Analysis,SSA)[40]、时态语义分析(Temporal Semantic Analysis,TSA)[41]、NASARI[42]、语境语义分析(Context Semantic Analysis,CSA)[43]方法等。WikiWalk是将ESA方法与维基百科网页面上所使用的个性化PageRank算法相结合,该方法相比于基准方法ESA在效果上有一些提升,但是计算成本太高。SSA和TSA方法是针对语义相关性问题而提出的方法,SSA方法是将维

基百科网页中带有链接的词语看作显著概念,然后每个词语由一组显著概念组成,通过生成每个词语的语义概要(semantic profiles)再计算概要的重叠程度来表示语义的相关程度。在TSA方法中,每个显著概念被表示为文档语料中的时间序列,这是第一次将时间信息引入到语义关联模型中的方法,而且作者通过实验证明该方法比ESA方法取得了更好的效果。NASARI方法与ESA方法类似,但是它结合了WordNet知识信息,改进了ESA方法中的降维方法和加权方案,得到了更有效的向量表示,对比ESA方法效果非常突出。由于上述方法专为使用维基百科作为知识来源而设计的,不能移植到一般的知识库中,Benedetti等人为解决这一问题又提出了CSA方法,该方法主要针对文档之间的相似度问题。

目前国际上关于基于网络知识的研究除了使用维基百科外还会使用DBpedia,它从维基百科的词条里撷取出结构化的信息,强化了维基百科的链接功能,减少了维基百科中的数据噪声和冗余。DBpedia也被认为是世界上最大的本体知识库,但它不同于传统的本体,它能根据维基百科数据的变化实现内容更新,也不需要具备严格的分层结构,现逐渐被越来越多的学者所关注。基于DBpedia的语义相似度计算方法的代表算法包括LDSD[44]和Shakti[45],目前一些改进实验的对比工作都会以上述两种算法作为基准。Piao等人通过实验证明LDSD方法优于Shakti方法,并对LDSD进行改进提出了Resim方法[46]。

对于中文的文本相似度计算,除了使用中文维基百科作为网络知识外,还有部分学者使用的是百度百科,其原理与基于维基百科的方法不尽相同,这方面的计算方法主要由中国学者提出,研究内容还非常少,主要包括以下两个内容:一是詹志建等人提出通过分析百度百科词条信息,从表征词条的解释内容方面综合分析词条相似度,通过计算部分之间的相似度得到整体的相似度[47]。二是尹坤等人提出将SimRank算法应用到百度百科的词条上,通过词条之间的链接关系计算词条语义相似度[48]。由此可见,相较于基于维基百科的众多研究方法,基于百度百科的方法还有待进一步的探索。

网络知识虽然有丰富的语义信息、更新迅速的信息内容,但是其数据噪声很大,与WordNet等语义词典相比,其数据的结构性也不强,因此计算结果普遍较差,但是并不失为一个很有前景的研究领域。值得一提的是,使用基于网络知识进行语义相关度计算的效果相对很好,国内关于这方面的研究多于对语义相似度的计算。这主要由于语义相关度包含了同义关系、反义关系、上下位关系等,相较于对语义相似度的计算要求并不严格,网络知识库中的链接关系也恰好能够很好地体现文本之间的相关性。

3.2 基于语料库

基于语料库的语义相似度计算,是根据从大型语料库获得的信息确定两个文本之间的相似性,其中语料库是用于语言研究的大量书面语或口语的文本集合,可以根据任务的领域不同有选择对语料库进行选取,如比较常用的维基百科语料、百度百科语料,针对特定领域的文学语料、新闻语料、金融语料等,针对口语的知乎语料、微博语料等,如果待处理任务本身的文本足够大,也可以将这些文本的集合作为语料。Li等人在2018年的论文中,对中文文本处理中常用的几种语料进行了整理,这是目前较新的中文语料信息[49]。

基于语料库的方法可以分为两类,一类是基于分布表示(distributional representation)[50],主要是利用语料库将文本转化为具有语义信息的向量表示形式,可以根据向量相似度来判断语义/分布相似度,或作为机器学习算法的特征,也可以输入到神经网络中进行学习。这种方法是基于语料库方法中的一个大类,也是目前研究最为主流的方向。另一类是基于搜索引擎(Web Search Engines-Based)的方法,将整个Web看作是一个动态语料库,研究工作侧重于计算词之间的语义相似度,目前的研究热度虽不如前者,但是该方法在语义相似度计算发展史中有着举足轻重的地位。基于语料库的整体分类体系如图4所示。

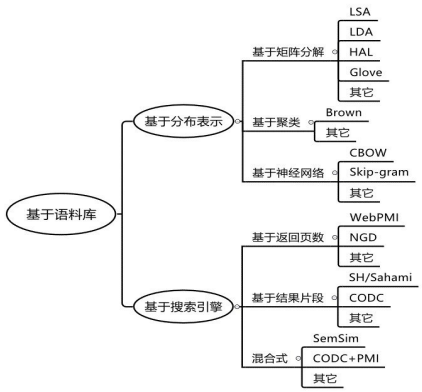

图4 基于语料库的语义相似度计算方法分类

3.2.1 基于分布表示

基于分布表示的模型是通过在对词语发生的上下文进行统计分析,动态地构建语义表示。其基本思想来源于1954年Harris提出的分布假说(distributional hypothesis):上下文相似的词具有相似的语义。基于分布假说得到的表示均可以称为分布表示[50]。与这种表示方式相对的是在表面文本相似度计算方法中所提到的one-hot向量表示和TF-IDF向量表示,这类表示方法只是将词符号化,而不包含任何的语义信息。基于分布假说的模型的词表示方法,根据建模方式的不同主要分为三类:基于矩阵的分布表示、基于神经网络的分布表示和基于聚类的分布表示[50-51]。由于基于矩阵的分布表示方法出现的时间最早,算法也最丰富,相当长的时间里基于矩阵的分布表示直接被表述为分布表示或分布语义模型(Distributional Semantic Models,DSMs),目前部分学者仍会使用这样的表述方式[52],同时也有一些学者将整个基于分布假说的模型称为分布语义模型[51,53]。除此之外,分布式表示(distributed representation)、词嵌入(word embedding)被一些学者特指基于神经网络的分布表示[50],也有学者

将三类表示统称为分布式表示[54-55]。部分学者也将基于矩阵的分布表示表述为计数模型或共现模型,将基于神经网络的模型表述为预测模型[53]。目前关于这些术语的使用界限尚未有非常明确和权威的说明。

基于矩阵的分布表示需要构建一个"词-上下文"矩阵,从矩阵中获取词的表示,在矩阵中,每行对应一个词,每列表示一种不同的上下文,矩阵中每个元素对应相应词和上下文的共现次数,矩阵中的一行即为对应词的表示,这种表示描述了该词的上下文分布。上下文可以是目标词所在的文档、目标词附近的若干个词或目标词附近各词组成的N元词组(N-gram),使用的上下文不同构成的矩阵稀疏程度也不同,其中"词-文档"矩阵最为稀疏,计算效果也最差。矩阵中的元素值除了使用目标词与上下文的共现次数,许多学者还会使用TF-IDF、PMI或取对数,以实现对元素值的加权或平滑。最后使用矩阵分解技术(可选)如奇异值分解(Singular Value Decomposition,SVD)、非负矩阵分解(Non-negative Matrix Factorization,NMF)、典型关联分析(Canonical Correlation Analysis,CCA)或Hellinger PCA(HPCA)等将原始的"词-上下文"矩阵从高维稀疏向量压缩为低维稠密向量[50]。

基于矩阵的分布表示方法在上述步骤的基础上衍生了非常多样的模型,比较经典的模型包括潜在语义分析(Latent Semantic Analysis,LSA)[56]和语义存储模型(Hyperspace Analogue to Language,HAL)[57]模型。LSA在信息检索领域也被称为潜在语义索引(Latent Semantic Indexing,LSI),它使用词所在的文档作为上下文形成"词-文档"矩阵,然后利用SVD进行矩阵降维,既可以去除数据中潜在的噪声,也使矩阵不再稀疏,向量更加平滑,最后通过取任意两行向量的夹角余弦作为其对应词语之间的相似度。HAL模型将目标词的邻近词语作为上下文,假定语料中含有N个词就可以构成一个的N*N的矩阵,矩阵元素是对应行和列的词语之间的关联强度,另外可以选择将低熵列从矩阵中删除。HAL只需要语料信息不需要人工指定维数,可以通过对共现(co-occurrence)信息的处理方式不同来决定是否记录语序信息。

在上述方法的基础上,又相继出现了广义潜在语义分析(Generalized Latent Semantic Analysis,GLSA)[58],概率潜在语义分析(Probabilistic Latent Semantic Analysis,PLSA)[59]、DISCO(Extracting DIStributionally similar words using CO-occurrences)[60]以及各种主题模型的变体等方法。由于基于矩阵的分布表示有着悠远而丰富的发展历程,上述几种方法还只是该类方法中的一小部分。Baroni等人在2014年的研究中通过大量的对比实验证明使用PMI加权方案和SVD降维技术组合得到的PMI-SVD方法效果最优[53]。最新的研究成果Glove[61]模型是对"词-词"矩阵进行分解得到词表示方法,矩阵元素是对应词语在语料中的共现次数的对数,矩阵分解技术借鉴了推荐系统中使用的基于隐因子分解(Latent Factor Model)的方法。在计算重构误差时只考虑共现次数非零的矩阵元素,同时对矩阵中的行和列加入偏移项。有大量的学者通过实验证明了该方法在整体计算效果上优于传统的基于矩阵的分布表示方法,成为了目前较为流行的词向量表示方法之一[62]。

基于神经网络的分布表示通常指由神经网络模型对文本进行词粒度的分析得到的低维实数向量表示,主要针对上下文和上下文与目标词之间的关系建模,该方法最大的优势在于可以表示复杂的上下文。基于神经网络的分布表示模型最经典的模型为CBOW和Skip-gram,由Mikolov等人在2013年同时提出,统称为Word2vec[63],同年作者对模型做出了进一步的改进,并提出将文本中的固定习语提取出后进行整体的向量表示可以获得更优的效果。目前这两个模型被工业界和学术界广泛使用[28]。Doc2vec模型是Word2vec模型的扩展,用来学习任意长度文本的分布向量表示,该模型包括两种结构,分别是Distributed Memory version of Paragraph Vector (PV-DM)和Distributed Bag of Words version of Paragraph Vector (PV-DBOW)[64]。PV-DM模型将每个段落表示成一个向量,通过这个段落和段落中的连续的几个词来预测下一个词。而在PV-DBOW模型中,只使用段落向量预测当前段落中特定大小窗口中的词。

基于聚类的分布表示是通过聚类手段构建词和上下文之间的关系,最经典的方法是布朗聚类(Brown Clustering)[65]。布朗聚类是一种层级聚类方法,聚类结果为每个词的多层类别体系,根据两个词的公共类别判断这两个词的语义相似度,通过对词进行聚类可以最大限度地利用互信息。布朗聚类只考虑相邻词之间的关系,每个词只使用它上一个词作为上下文信息,因此也忽略了词语在全部语料中的使用情况。

针对中文文本的特殊性,中国学者Chen等人在2015年提出了基于汉字的CWE(character- enhanced word embeddding model)模型,该模型是在Word2vec中CBOW基础上进行改进,思路是在词向量表示中融入单个汉字的向量信息[66]。Yu等人在2017年提出了JWE(Joint Learning Word Embedding)模型,该模型将汉字拆分成多个的字件,然后将词向量、字向量和字件向量三种向量表示信息融合在一起[67]。这两种模型是中文文本词向量表示发展史中比较有代表性的两种方法。另外,蚂蚁金服在2018年的AAAI会议上公开了一种基于汉字笔画信息的中文词向量表示算法,为中文的词向量分布表示提供了新思路,并通过实验对比证明了该方法在词语相似度计算上优于其他方法[68]。

尽管不同的分布表示方法使用了不同的技术手段获取文本表示,但由于这些方法均基于分布假说,它们的核心思想都由两部分组成:一是选择一种方式描述上下文;二是选择一种模型刻画目标词与其上下文的关系。在这三类模型中,基于聚类的模型较为特别,是将词表示为聚类的类标,如果采用层级聚类方法,可以根据聚类类别的公共前缀衡量词之间的相似度[50],目前使用这类方法进行文本相似度计算的研究内容并不如另外两类丰富。另外两类模型得到的都是向量表示,可以直接使用在表面文本相似度计算方法一节中提到的余弦距离、欧氏距离等向量空间距离衡量指标来计算文本的相似度,Levy等人在2014年的论文中证明了

基于矩阵的分布表示和基于神经网络的方法在一定程度上是等价的[69]。

基于分布表示的语义相似度计算因为需要利用语料中的信息将文本转化为向量，所以这种方法比较依赖于训练时所用的语料库，其计算结果受数据稀疏和数据噪声的干扰非常大。而且这种方法相对于其它方法计算量大，计算方法也比较复杂。但是使用该方法得到的向量表示可以重复利用，其计算效果也非常可观，已成为目前使用最多、研究最为广泛的一类方法。

3.2.2 基于搜索引擎

搜索引擎为海量信息提供了高效的接口，查询页面数和查询结果片段是大多数搜索引擎提供的两个有用的信息源[70]。基于搜索引擎的语义相似度计算能够适应层出不穷的新词汇，受到许多研究者的关注[71]。其算法通常采用基于查询返回页数(page counts-based)、基于查询结果片段(text snippets-based)，或将两种方式结合来进行语义相似度的计算。

一个查询词P的返回页数是指在搜索引擎中查询词语P时返回的包含查询词P的网页总数 $N(p)$ 。对于连接查询词P和词Q的返回页数可以看作是对词语P和Q共现的全局度量，使用 $N(p \cap q)$ 来表示，这些值通常表示的是实际值的估计值。基于查询返回页数的语义相似度计算的方法有很多，最具代表性的方法是Cilibrasi等人于2007年提出的归一化谷歌距离(Normalized Google Distance，NGD)[72]。同年，Bollegala等人启发式地将四种常见的文本共现方法Jaccard、Overlap、Dice、Pointwise mutual information(PMI)应用在谷歌搜索引擎的查询返回页数上，并通过对比实验证明PMI的计算效果最好[73]。张硕望等人针对百度搜索引擎将WebPMI修改为PMIB，使得算法更适用于中文词汇相似度计算[34]，吴克介等人将归一化谷歌距离应用在百度搜索引擎中并与PMIB算法相结合，提出了优化的基于百度搜索引擎的PMINB算法[71]。上述几种方法的具体公式如表2所示，其中N表示一个搜索引擎包含的全部网页数，也称索引总数。

使用返回页数进行语义相似度计算有两大优势：一是该方法使用的唯一信息是搜索引擎在不到1秒内即可返回的页数，二是该方法不依赖于任何领域信息，即使在人类感知中没有任何关联的词语依然能计算相似度[74]。但这种方法的缺点也很明显：首先它忽略了词语在页面中的位置，即使一个页面中同时出现了两个词语，也不能断定两个词语是相关的。其次，多义词语的页数包含其所有意义的页数之和，例如苹果的页数包含了苹果作为水果的页数和苹果作为公司的页数。此外，由于网络中的噪音和冗余，有些词可能在某些页面上同时出现而没有实际关联，大量重复的页面使得计算结果不准确。基于上述原因，在度量语义相似性时，仅使用基于查询返回页数的方法是不可靠的[70]。

查询结果片段是搜索引擎在文档中的查询词周围提取的简短文本窗口，提供了关于查询术语的上下文信息。片段对于搜索非常有用，因为用户可以通过读取片段来决定搜索结果是否相关，而不必打开链接。将片段看作上下文可以避免从Web中直接下载源文档进而提高计算效率[75]。基于查询结果片段的方法中比较典型的算法包括Sahami等人提出的SH(Sahami and Heilman)方法[76]和Chen等人提出相关性双重检测(Co-occurrence Double-Checking，CODC)算法[77]。SH方法是利用搜索引擎收集每个查询的返回片段，并将各个片段转化为基于TF-IDF权重的向量表示形式，通过归一化得到片段的质心向量，然后将两个查询词的语义相似度定义为对应质心向量之间的内积。CODC算法是在搜索引擎中分别收集词语P和Q的文本片段，计算词语P的片段中出现Q的次数和Q的片段中出现P的次数，然后将两个数值进行非线性组合后作为相似度计算的结果值。Bollegala等人通过大量的对比实验证明基于查询结果片段的方法普遍优于基于查询返回页数的方法[73]，但该方法有一个公认的缺点，由于Web的庞大规模和与查询相关的文档数量之多，只有查询结果排名靠前的一些片段会被有效的处理。查询结果的排名是由搜索引擎特有的各种因素的复杂组合决定的，因此不能保证我们需要度量语义相似度的词对的信息包含在排名靠前的文本片段中，这一弊端导致这种方法的语义相似度计算结果可能出现零值。相较于上述方法更为传统的方法还包括点互信息检索(Pointwise Mutual Information - Information Retrieval，PMI-IR)[78]，PMI-IR是基于AltaVista搜索引擎查询的方法，该方法曾在大量的论文中出现过，但是现已不在AltaVista上使用。

为克服基于返回页数和基于查询结果片段这两类方法存在的问题，研究学者提出将两种方法结合起来进行相似性的度量。例如，Bollegala等人提出一种浅层词法模式抽取的方法来捕捉文本片段中词语之间的语义关系，将各个模式的频率值与几种基于返回页数方法的度量值相结合作为特征，

表2 基于查询返回页数的语义相似度计算方法

方法	定义
Jaccard	$W_{jacc}(p,q)=N(p\cap q)/N(p\cup q)$
Overlap	$W_{over}(p,q)=N(p\cap q)/\min(N(p),N(q))$
Dice	$W_{dice}(p,q)=2*N(p\cap q)/(N(p)+N(q))$
PMI	$W_{pmi}(p,q)=\log_2(\frac{N*N(p\cap q)}{N(p)*N(q)})/\log_2 N$
NGD	$NGD(p,q)=\frac{\max(\log N(p),\log N(q))-\log N(p\cap q)}{\log N-\min(\log N(p),\log N(q))}$
PMIB	$PMIB(p,q)=\log(\frac{N_b*N(p\cap q)}{N(p)+N(q)})/\log N_b$
PMINB	$PMINB(p,q)=\beta*PMIB(p,q)+(1-\beta)(1-NGD(p,q))$

使用支持向量机对词对进行同义和非同义的分类，此方法被命名为SemSim[73]，作者于2011年又对它进行了改进，使计算效率和准确率得到进一步的提升[70]。高国强[79]等人同时分析CODC和PMI两种方法，提出根据不同情况使用不同的算法，如果两个词的语义相关性较强则使用CODC算法，否则使用PMI算法，这在一定程度上减轻了CODC和PMI两种算法各自的局限性，增加了结果的可信度。陈海燕[80]利用Google搜索独有的去冗余方法，整合查询返回页数、结果片段和重复记录数三种信息来修改PMI算法，效果比较明显，但中英文之间的差异和搜索引擎算法之间的差异使得该方法不适用于中文语义相似度计算。张硕望[34]等人针对中文词汇，提出将PMI算法与CODC算法应用于百度搜索引擎，利用批量梯度下降法学习权值参数，融合《知网》与搜索引擎完成词汇语义相似度计算。

基于搜索引擎的语义相似度计算不需要先验知识和本体，是语义相似度计算方法中计算效率最高、原理最为简单的一类方法，但是该方法受限于网络数据上的噪声和冗余，其计算准确率在整体上并不如其他的语义相似度算法。而且研究工作大多是对前人工作的改进，还只是停留在词汇之间的语义相似度计算，针对句子和文档的研究非常少，主要因为将长文本作为查询发送到搜索引擎可能会返回很少的结果甚至没有。

4 结 语

文本相似度计算是自然语言处理中的一项基础性研究，有着非常悠久的研究历史，近年来随着神经网络的出现，文本相似度计算的准确度更是得到了显著提升。本文简要总结了以往研究中的经典方法，并且对当前主流的研究方法进行总结分析。通过对文本相似度计算中大量的方法进行梳理和分析，主要将其分为表面文本相似度计算和语义相似度计算两方面来，其中，表面文本相似度计算主要包括基于字符和基于术语的计算方法，语义相似度计算可以分为基于知识库和语料库两大类计算方法，基于知识库的计算方法包括基于本体知识和网络知识两种，基于语料库的计算方法包括基于分布表示和基于搜索引擎两种，未来基于神经网络的分布表示方法是文本相似度计算领域最为重要的研究方向。本文对国内外学者取得的进展和最新成果进行了总结归纳，更正了以往文献中分类杂乱和交叉的内容，最终形成了较为全面的分类体系，希望有助于全面把握和深入了解文本相似度计算方法的研究现状和未来趋势。

参考文献

1 张金鹏.基于语义的文本相似度算法研究和应用[D].重庆:重庆理工大学,2014.

2 Gomaa W H, Fahmy A A. Short Answer Grading Using String Similarity And Corpus-Based Similarity [J]. International Journal of Advanced Computer Science and Applications,2012,3(11):114-121.

3 Kadupitiya J , Ranathunga S , Dias G . Short Sentence Similarity Calculation using Corpus-Based and Knowledge-Based Similarity Measures[C]// Proceedings of the 26th International Conference on Computational Linguistics, Osaka,2016.

4 Gomaa W H, Fahmy A A. A Survey of Text Similarity Approaches [J]. International Journal of Computer Applications, 2013, 68(13): 13-18.

5 陈二静,姜恩波.文本相似度计算方法研究综述 [J].数据分析与知识发现,2017,(6):1-11.

6 Levenshtein, Vladimir I. Binary codes capable of correcting deletions, insertions, and reversals [J]. Soviet Physics Doklady, 1966,10 (8): 707 - 710.

7 Melamed I D. Automatic evaluation and uniform filter cascades for inducing n-best translation lexicons [C]//Proceedings of the 3rd Workshop on Very Large Corpora,Engish,1995:184-198.

8 张焕炯,王国胜,钟义信.基于汉明距离的文本相似度计算[J].计算机工程与应用,2001,(19):21-22.

9 Kondrak. N-Gram Similarity and Distance [J]. String Processing and Information Retrieval, Lecture Notes in Computer Science, 2005,(3772): 115-126.

10 Bard Gregory V. Spelling-error tolerant, order-independent pass-phrases via the Damerau - Levenshtein string-edit distance metric [C]// Proceedings of the Fifth Australasian Symposium on ACSW Frontiers , Ballarat,2007:117-124.

11 Winkler W E. String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage [C]//Proceedings of the Section on Survey Research Methods, American Statistical Association, Alexandria, January,1990:354-359.

12 Needleman B S, Wunsch D C. A general method applicable to the search for similarities in the amino acid sequence of two proteins [J]. Journal of Molecular Biology,1970,48 (3): 443 - 53.

13 Smith F T, Waterman S M. Identification of Common Molecular Subsequences [J]. Journal of Molecular Biology,1981, (147): 195-197.

14 Salton G, Wong A, Yang C S. A Vector Space Model for Automatic Indexing [J]. Communications of the ACM, 1975,18(11): 613-620.

15 Jaccard P. Étude comparative de la distribution florale dans une portion des Alpes et des Jura [J]. Bulletin de la Société Vaudoise des Sciences Naturelles,1901,(37):547-579.

16 Dice, L. Measures of the amount of ecologic association between species [J]. Ecology, 1945,26(3):297-302.

17 Eugene F K. Taxicab Geometry[M]. US:Dover Publica-

tions, Dover,1987.

18 Bray J R, J T Curtis. An ordination of upland forest communities of southern Wisconsin[J]. Ecological Monographs, 1957,(27):325-349.

19 Jiang Zhao, Tian Tian Zhu, Man Lan. ECNU: One stone two birds: Ensemble of heterogenous measures for se-mantic relatedness and textual entailment [C]//Proceedings of the 8th International Workshop on Semantic Evaluation, Dublin,2014:271-277.

20 Anshumali Shrivastava, Ping Li. In Defense of MinHash Over SimHash [J]. Eprint Arxiv,2014,7(3):886-894.

21 Charikar M S. Similarity Estimation Techniques from Rounding Algorithms[C]//Proceeding of the 34th annual ACM Symposium on theory of computing, Montreal, 2002: 380-388.

22 王 源. 一种基于simhash的文本快速去重算法[D]. 长春: 吉林大学,2014.

23 Henzinger, Monika. Finding near-duplicate web pages: a large-scale evaluation of algorithms [C]//Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle,2006.

24 Manku G S, Jain A, Das S A. Detecting near-duplicates for web crawling [C]//Proceedings of the 16th International Conference on World Wide Web,Banff,2007:141-149.

25 Das A S, Datar M, Garg A,et al. Google news personalization: scalable online collaborative filtering [C]//Proceedings of the 16th International Conference on World Wide Web, Banff, 2007:271-280.

26 Lin C Y. Rouge: a package for automatic evaluation of summaries [C]//Proceedings of the Workshop on Text Summarization Branches Out, Barcelona,2004.

27 Papineni K, Roukos S, Ward T, et al. BLEU: a method for automatic evaluation of machine translation [C]//Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Philadelphia,2002:311 - 318.

28 Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and their Compositionality [J].Advances in Neural Information Processing Systems, 2013,(26):3111-3119.

29 Mihalcea R, Corley C, Strapparava C. Corpus-based and Knowledge-based Measures of Text Semantic Similarity [J]. National Conference on Artificial Intelligence & the Eighteenth Innovative Applications of Artificial Intelligence Conference, 2006,20(1):775-780.

30 陈宏朝,李 飞,朱新华,马润聪. 基于路径和深度的同义词词林词语相似度计算[J]. 中文信息学报,2016,5(30):80-88.

31 刘宏哲,须 德. 基于本体的语义相似度和相关度计算研究综述[J]. 计算机科学,2012,2(39):9-13.

32 Dr B Poorna, A Sudha Ramkumar. Semantic Similarity Measures: an Overview and Comparison [J]. International Journal of Advanced Research in Computer Science,2018,1 (9):100-103.

33 Althobaiti AFS. Comparison of Ontology-Based Semantic-Similarity Measures in the Biomedical Text [J]. Journal of Computer and Communications, 2017,(5):17-27.

34 张硕望,欧阳纯萍,阳小华,等. 融合《知网》和搜索引擎的词汇语义相似度计算[J]. 计算机应用,2017,37(4):1057-1060.

35 闫 红,李付学,周 云. 基于HowNet句子相似度的计算[J]. 计算机技术与发展,2015,25(11):53-57.

36 Strube M, Ponzetto S P. WikiRelate! Computing Semantic Relatedness Using Wikipedia [C]//Proceedings of the 21th National Conference on Artificial Intelligence,Boston,2006: 1419-1424.

37 Gabrilovich E, Markovitch S. Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis [C]//Proceedings of the 20th International Joint Conference on Artifical Intelligence, Hyderabad,2007:1606-1611.

38 Milne D, Witten I H. An Effective, Low-cost Measure of Semantic Relatedness Obtained from Wikipedia Links[C]// Proceedings of the 23rd Association for the Advancement of Artificial Intelligence, Chicago, 2008.

39 Yeh E, Ramage D,Manning C D, et al. Wikiwalk: random walks on wikipedia for semantic relatedness[C]// Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing, Suntec,2009:41 - 49.

40 Hassan S, Mihalcea R. Semantic relatedness using salient semantic analysis[C]//Proceedings of the Twenty- Fifth AAAI Conference on Artificial Intelligence, Urbana, 2011.

41 Radinsky K, Agichtein E, Gabrilovich E,et al. A Word at a Time: Computing Word Relatedness Using Temporal Semantic Analysis[C]//Proceedings of the 20th International Conference on World Wide Web, Hyderabad, 2011:337-346.

42 Camacho Collados J, Pilehvar M T, Navigli R. NASARI: a Novel Approach to a Semantically-Aware Representation of Items[C]//Proceedings of the North American Chapter of the Association of Computational Linguistics, Denver, Colorado, 2015:567-577.

43 Benedetti F, Beneventano D, Bergamaschi S. Context Semantic Analysis: a knowledge-based technique for computing inter- document similarity[C]//Similarity Search and Applications : 9th International Conference, Tokyo,2016.

44 Passant A. Measuring Semantic Distance on Linking Data and Using it for Resources Recommendations[C]//AAAI Spring Symposium: Linked Data Meets Artificial Intelligence, Menlo Park, Atlanta, Georgia, USA, 2010:93-98.

45 Leal J P, Queiros R. Computing semantic relatedness using dbpedia[C]//1st Symposium on Languages, Applications and Technologies, Braga, 2012:133-147.

46 Piao G, Ara S S, Breslin J G. Computing the Semantic Similarity of Resources in DBpedia for Recommendation Purposes[J].Springer International Publishing,2015,(7):185-200.

47 詹志建,梁丽娜,杨小平. 基于百度百科的词语相似度计算[J]. 计算机科学,2013,(6):199-202.

48 尹 坤,尹红风,杨 燕.基于SimRank的百度百科词条语义相似度计算[J]. 山东大学学报,2014,(3):29-35.

49 Shen Li, Zhe Zhao. Analogical Reasoning on Chinese Morphological and Semantic Relations[C]// Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics(ACL), Melbourne,2018.

50 来斯惟.基于神经网络的词和文档语义向量表示方法研究[D]. 北京:中国科学院大学,2016.

51 Joseph Turian, Lev Ratinov, Yoshua Bengio. Word representations: a simple and general method for semi-supervised learning[C]// Proceedings of the 48th annual meeting of the association for computational linguistics (ACL), Upsala, 2010:384 - 394.

52 Evert S. Distributional Semantic in R with the wordspace package[C]//Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: System Demonstrations, Dublin, Ireland, 2014: 110-114.

53 Baroni M, Dinu G, Kruszewski G. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors[C]//Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore,2014:238 - 247.

54 Dumancic S, Blockeel H. Clustering-Based Relational Unsupervised Representation Learning with an Explicit Distributed Representation[C]//Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, Melbourne, 2017:1631-1637.

55 Posadas-Durán J-P, Gómez-Adorno H, et al. Application of the Distributed Document Representation in the Authorship Attribution Task for Small Corpora Preprint version[J]. Soft Computing,2017,21 (3) :627-639.

56 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.

57 Lund K, Burgess C, Atchley R A. Semantic and associative priming in a high-dimensional semantic space[J]. Cognitive Science Proceedings (LEA), 1995,(4):660-665.

58 Matveeva I. Term representation with Generalized Latent Semantic Analysis[C]//Proceeding of recent advances in nature language processing (RANLP), Borovets, 2007:45.

59 Hofmann Thomas. Probabilistic Latent Semantic Indexing [C]//Proceedings of the Twenty-Second Annual International SIGIR Conference on Research and Development in Information Retrieval,Berkley, 1999:50-57.

60 Peter Kolb. Experiments on the difference between semantic similarity and relatedness[C]// The 13st Nordic Conference on Computational Linguistics (NODALIDA), Odense Denmark, 2009:81 - 88.

61 Jeffrey P, Richard S, Christopher D M. GloVe:Global Vectors for Word Representation[C]// Proceedings of the Empirical Methods in Natural Language Processing, Doha, 2014:1532-1543.

62 Levy O, Goldberg Y, Dagan I. Improving distributional similarity with lessons learned from word embeddings[J]. Bulletin De La Société Botanique De France, 2015,75 (3): 552-555.

63 Tomas Mikolov, Kai Chen, Greg Corrado, et al. Efficient estimation of word representations in vector space[C]// International Conference on Learning Representations Workshop Track, Scottsdale, 2013:1-12.

64 Le Quoc V, Mikolov Tomas. Distributed Representations of Sentences and Documents [C]//Proceedings of the 31st International Conference on Machine Learning,2014:1-9.

65 PF Brown, PV Desouza, RL Mercer. Class-based n-gram models of natural language[J]. Computational linguistics, 1992,18(4):467 - 479.

66 Chen X, Xu L, Liu Z, et al. Joint Learning of Character and Word Embeddings[C]//Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, Buenos Aires,2015:1236-1242.

67 Jinxing Yu, Xun Jian, Hao Xin. Joint Embeddings of Chinese Words, Characters, and Fine-grained Subcharacter Components[C]//Conference on Empirical Methods in Natural Language Processing, Copenhagen, 2017:286-291.

68 Cao S, Lu W, Zhou J, et al. cw2vec: Learning Chinese Word Embeddings with Stroke n-grams[C]//Association for the Advancement of Artificial Intelligence 2018, New Orleans, 2018.

69 Levy O, Goldberg Y. Neural Word Embedding as Implicit Matrix Factorization[J]. Advances in Neural Information Processing Systems (NIPS), 2014,(6):2177 - 2185.

70 Bollegala D, Matsuo Y, Ishizuka M. A Web Search Engine-Based Approach to Measure Semantic Similarity between Words[J]. IEEE Transactions on Knowledge & Data Engine, 2011,23 (7) :977-990.

71 吴克介,王家伟.基于知网和搜索引擎的词汇语义相似度计算[J]. 计算机与现代化,2018,(4):90-94.

72 Cilibrasi R L, Vitanyi P M B. The Google similarity distance[J]. IEEE Transactions on Knowledge and Data Engineering,2007,19(3):370-383.

73 Bollegala D, Matsuo Y, Ishizuka M. Measuring Semantic Similarity between Words Using Web Search Engines[C] // Proceedings of 16th International World Wide Web Conference, Banff,2007:757-766.
74 Song X. Ontology-based Domain-specific Semantic Similarity Analysis and Applications[D]. South Carolina:Clemson University, South Carolina,2018.
75 Shweta A Koparde. Measuring Semantic Similarity between Words Using Web Search Engines:A Survey[J].National Symposium on engineering and Research,2015,5(2):52-54.
76 Sahami M, Heilman T. A web-based kernel function for measuring the similarity of short text snippets[C]//Proceeding of 15th International World Wide Web Conference, Edinburgh, 2006.
77 Chen H H, Lin M S, Wei Y C. Novel association measures using Web search with double checking [C]//Sydney:Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics, 2006:1009-1016.
78 Turney P D.Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL[J]. Springer Berlin Heidelberg,2001, (2167):491-502.
79 高国强,黄吕威,陈丰钰.使用网络搜索引擎计算汉语词汇的语义相似度[J].计算机技术与发展,2014,24(7):84-87.
80 陈海燕.基于搜索引擎的词汇语义相似度计算方法[J].计算机科学,2015,42(1):261-267.

(责任编辑:徐 波)

(上接第151页)

6 史 波. 网络舆情群体极化的动力机制与调控策略研究[J].情报杂志, 2010, 29(7): 50-53,69.
7 霍凤宁, 禹婷婷, 孙宝文.网络群体极化的判定、测量与干预策略研究 [J].电子政务, 2015,(10): 19-26.
8 Changjun L, Jieun S, Ahreum H. Does social media use really make people politically polarized? Direct and indirect effects of social media use on political polarization in South Korea [J]. Telematics and Informatics, 2018,(35): 245-254.
9 Sounman H, Sun H K. Political polarization on twitter: Implications for the use of social media in digital governments [J].Government Information Quarterly, 2016,(33): 777-782.
10 Jessica K, Leaf V B, Charles M J. Partisan underestimation of the polarizing influence of group discussion [J]. Journal of Experimental Social Psychology, 2016,(65):52-58.
11 Quansheng W, Xue Y, Wanyu X. Effects of group arguments on rumor belief and transmission in online communities: An information cascade and group polarization perspective[J]. Information & Management, 2018,(55):441-449.
12 王益成, 王 萍, 王美月, 张卫东. 信息运动视角下内容平台突破"信息茧房"策略研究 [J]. 情报理论与实践, 2018, 41(5): 114-119.
13. [美]赛琪·莫斯科维奇.流氓的时代 [M]. 南京: 江苏人民出版社, 2003: 22.
14 闫泽华. 内容算法——把内容变成价值的效率系统 [M]. 北京: 中信出版集团, 2018:39.
15 [法]古斯塔夫·勒庞.乌合之众: 大众心理研究 [M]. 北京: 中央编译出版社, 2005.
16 Wallach M A, Kogan N, Bem D J. Diffusion of Responsibility and Level of Risk Taking in Groups [J].Journal of Abnormal and Social Psychology, 1964,(68):263-274.

(责任编辑:徐 波)