基于 NLP 的文本相似度检测方法

代晓丽[1,2]，刘世峰[1]，宫大庆[1]
（1. 北京交通大学经济管理学院，北京 100044；2. 北京信通传媒有限责任公司，北京 100078）

摘 要：针对当前的文本相似度检测方法忽略文档结构信息、缺乏语义关联性的问题，提出了面向文本的相似度检测方法。首先，采用层次分析法（AHP）计算词语位置权重以提取特征词。其次，引入 Pearson 相关系数度量词语间的语义关联，并将其作为广义 Dice 系数的权重计算相似度。实验表明，所提方法在提高特征词提取的精确度、相似度计算结果的准确率方面表现良好。

关键词：文本相似度；词语位置权重；层次分析法；特征词提取；Pearson 相关系数
中图分类号：TP391
文献标识码：A
DOI: 10.11959/j.issn.1000−436x.2021192

Text similarity detection method based on NLP

DAI Xiaoli[1,2], LIU Shifeng[1], GONG Daqing[1]
1. School of Economics and Management, Beijing Jiaotong University, Beijing 100044, China
2. China InfoCom Media Group, Beijing 100078, China

Abstract: Current text similarity detection methods that ignore document structure information and lack semantic relevance. To solve these problems, a text-oriented similarity detection method was proposed. First, analytic hierarchy process (AHP) was used to calculate word position weight to extract feature words. Second, the Pearson correlation coefficient was used to measure semantic correlation between words which was the weight of generalized Dice coefficient to calculate similarity. Experimental results show that the proposed method can improve the precision of feature word extraction and the accuracy of similarity calculation results.

Keywords: text similarity, word position weight, analytic hierarchy process, feature word extraction, Pearson correlation coefficient

1 引言

互联网的发展给网络平台带来了海量的数据，其中文本数据是主要的数据形式，如何处理网络中的大量文本数据是一个急需解决且复杂的问题。文本相似度检测是文本处理领域的一个关键技术，通过文本间的对比计算两篇或多篇文本间的相似程度，在信息检索[1]、文本分类[2]、机器翻译[3]、自动问答[4]等自然语言处理（NLP, natural language processing）领域的任务中具有广泛应用。

由于文本格式、类型繁多，很难对文本的各种特征进行捕捉，使设计一个准确性较高的文本相似度检测方案面临一定的挑战。基于统计和基于语义的文本相似度检测方法是学者们研究的热点[5]。

基于统计的文本相似度检测方法主要是基于字符匹配和基于词频特征的相似比较，基于字符匹配的方法将文本分解为字的集合，以字符间的变化程度作为相似度结果，最长公共子串（LCS, longest common substring）[6]、编辑距离[7]、Jaccard 系数[8]、Dice 系数[9]等是较常用的方法；基于词频特征的方

收稿日期：2021−07−01；修回日期：2021−09−13
通信作者：宫大庆，dqgong@bjtu.edu.cn
基金项目：国家自然科学基金资助项目（No.J1824031）
Foundation Item: The National Natural Science Foundation of China (No.J1824031)

法以 TF-IDF（term frequency-inverse document frequency）方法为主，该方法将文本分解为词语的集合，以词频作为向量，通过计算向量距离得到文本相似度，如欧氏距离、曼哈顿距离、余弦距离[10]等。这些方法仅衡量了文本表面的相似度，而没有考虑文本的语义相似度，使得到的结果缺乏一定的准确性。

针对语义缺失的问题，出现了基于语义的方法，该方法通过引入外部知识来使文本具有语义信息[11]，其中基于词典和基于向量空间模型（VSM, vector space model）是较常见的方法。基于词典的方法利用通用词典构建词语的概念语义树，两词语在树中的距离即为它们之间的相似度[12]；基于向量空间模型的方法利用外部语料库来构建具有语义的词向量，通过度量词语的重要性提取特征词来表示文本，然后将特征词向量综合表示为文本向量，最后以文本向量间的距离作为相似度结果[13]。在基于向量空间模型的方法中，对于特征词提取，多数方法只依据词语的词频信息，没有考虑文本的结构信息；同时，文本向量表示没有考虑词语间的语义关联性，导致相似度检测结果的准确率较低。

为了解决上述问题，本文提出了面向文本的相似度检测方案，基于文档结构特征将词语位置权重与词频权重作为特征词提取的依据，并将词语间的语义关系融入相似度计算的过程，在提升特征词提取精度的同时提高相似度计算的准确性。本文的主要贡献如下。

1) 针对特征词提取阶段词语位置加权方法主观性较强导致提取结果缺少代表性的情况，提出了基于层次分析法（AHP, analytic hierarchy process）的词语位置加权方法，利用成对比较法基于文本结构设置词语位置权重，提高了特征词提取结果的精确度。

2) 针对相似度计算阶段的文本向量表示法未考虑词语间语义关系导致计算结果不够准确的情况，提出了基于 Pearson 相关系数和广义 Dice 系数的相似度计算方法，利用相关系数衡量词语间语义关系，改进广义 Dice 系数公式，提高了相似度计算结果的准确性。

3) 对本文提出的面向文本的相似度检测方案与经典方法、未做出改进的原始方法在准确率、精确率、召回率、F1 值方面进行对比，实验结果显示，本文方案有效提高了相似度计算的准确率。

2 相关工作

关键词提取是从文本中提取出最能代表该文本信息的词语，文本相似度的计算与关键词的提取有着密切的关系，关键词提取的准确率间接地影响相似度计算结果。最常用的关键词提取技术有 TF-IDF[14]、线性判别分析（LDA, linear discriminant analysis）[15]、图模型[16]，许多学者在此基础上做出了改进。传统的 TF-IDF 算法只在处理不同类文本时效果较好，文献[17]在 TF-IDF 算法的基础上提出 TF-IWF（term frequency-inverse word frequency），将逆文档频率改为逆词语频率并设置词语位置权重，能够更好地处理语料库中同类型文本较多的情况以及利用词语位置信息，但文中词语位置权重为作者的主观设置，缺少客观性。LDA 利用词语的概率分布推测文档的主题概率，文献[18]结合 TF-IDF 和 LDA 算法，利用 LDA 提取的主题构建关键词词典，基于该词典采用 TF-IDF 算法从文章的摘要中提取最终的关键词用于文章分类，提高了文本分类的精度，但关键词之间没有语义关联。文献[19]提出了基于 LDA 和图模型的关键词挖掘方法，采用两级语义关联模型，将主题之间的语义关系与主题下词语之间的语义关系联系起来，并根据组合作用提取关键词，该方法提高了从文本中提取关键词的准确性，但计算的复杂度较高。

针对文本相似度计算，现有方法从降低复杂度和提高准确率方面进行研究。文献[20]利用哈希将文本转化为数字指纹，使用 Jaccard 系数来度量指纹间的相似值，适用于检测字符级改变的文本。文献[21]提出了基于 VSM 的相似度计算方法，利用特征项权重加权 TF-IDF，提高了相似度计算的精度。文献[22]使用 VSM 和 TF-IDF 加权模式以及哈希特征提取技术提高了大规模文本相似度计算的速度。这些方法生成的都是高维稀疏的向量且不包含文本语义。文献[23]提出了基于双向空间模型的相似度计算，分别利用维基百科的数据链接和构建依赖树来计算词语相似度和文本结构相似度，双向结合得到文本相似。文献[24]提出了一种基于 TF-IDF 和 LDA 的混合模型来计算文本相似度，能够利用文本本身包含的语义信息并反映文本关键词的权重，但 LDA 包含的文本语义较稀疏。文献[25]提出了一种结合 HowNet 语义知识词典和 VSM 的文本相似度计算方法，在词汇层面使用 HowNet 计算相似度避

免了语义信息丢失，在文本层面使用 VSM 计算相似度保证了表达信息的完整性，但是 HowNet 等已构建好的通用词典较少，更新慢、具有独立性，跨领域或新领域的应用效果较差。文献[26]结合 Word2vec 词向量转换技术，利用其语义分析能力构建优化的 LDA 模型，最后使用余弦相似度来计算文本相似度，充分表达了文本语义，理想地实现了对重复文本的语义分析，但其训练语料需要经过 Word2vec 模型转换为词向量，再将向量作为输入训练 LDA 模型，导致模型训练成本较高。

3　基于 NLP 的文本相似度检测框架

本节介绍了文本相似度检测框架以及各个步骤的具体方法，并分析了流程中存在的问题。

3.1　文本相似度检测框架

图 1 是本文结合基于向量空间模型和基于分布式表示方法[13,27-31]提出的文本相似度检测框架，利用分布式词向量将文本映射到向量空间中，以此计算文本在向量空间上的相似度。该框架是一种较通用的相似度计算流程，研究者常在其中的一个或多个步骤中进行研究改进，以提高检测性能。本文所使用的具体方法和步骤如下。

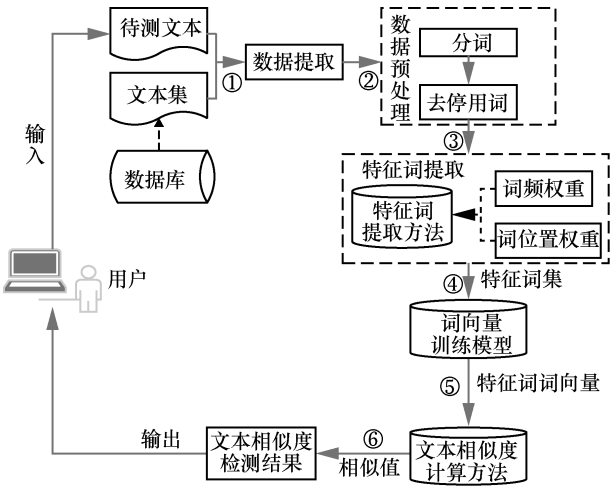

图 1　文本相似度检测框架

① 数据提取。用户将待测文本数据输入系统，系统从数据库中提取相应的文本集数据。

② 数据预处理。合并提取文本内容并对其分词、去停用词。首先使用分词工具将文本内容分割为词语集，由于词语集中会存在对文本表达无语义影响但会影响特征词提取结果的词语和符号，因此，使用停用词表，将这些词语和符号从词语集中删除。

③ 特征词提取。数据预处理结束后，接下来要从 2 个方面计算每个词语的总权重并作为特征词提取的依据。首先，使用 TF-IWF[17]算法计算词语的频率权重，如式(1)所示。

$$\text{TF-IWF}(i)=\text{TF}_i\times\text{IWF}_i=\frac{N_i}{N}\log\frac{P}{P_i} \tag{1}$$

其中，N_i 为词语 i 在单文本中的数量，N 为单文本词语总数，P 为语料库的词语总数，P_i 为词语 i 在语料库中的数量。这种方法能够有效降低文本集中文本数量少、同类型文本多等情况对词语权重的影响。其次，根据文本的结构特征设置词语的位置权重，表示为 Wloc(i)，对出现在文本标题、关键词、摘要中的词语分别赋予权值 3、2、1。最后，将词频权重和词位置权重加权和得到词语总权重，由大到小排序，提取一定比例的词语构成特征词集代表文本。词语 i 总权重计算式为

$$W(i)=0.5\text{TF-IWF}(i)+0.5\text{Wloc}(i) \tag{2}$$

④ 词向量生成。词向量生成是将词语转换为计算机可识别、可计算的过程。Word2vec 是一种词向量生成工具，由 Mikolov 等[32]于 2013 年开发，作为深度学习模型中的一种分布式表达。Word2vec 有 CBOW（continuous bag-of-words）和 Skip-gram（continuous skip-gram）2 种训练模式，CBOW 使用词语的上下文来预测词语本身，而 Skip-gram 则使用当前词来预测上下文词语。Word2vec 模型能够从大规模未经标注的语料中训练得到具有语义、低维、稠密的词向量，可以较好地应用于文本相似度中的词语表示。

⑤ 文本相似度计算。通过 Word2vec 模型得到特征词向量，利用式(3)的 2 种方式将其转换为文本向量，前者为叠加所有特征词向量，后者取叠加后词向量的平均值。然后使用广义 Dice 系数[33]计算 2 个文本向量的相似度表示为最终的文本相似度，如式(4)所示。

$$d_t=\sum_{x=1}^{n}k_x=\frac{\sum_{x=1}^{n}k_x}{n} \tag{3}$$

$$\text{sim}(d_i,d_j)=\frac{2\sum_{i=1}^{n}x_iy_i}{\sum_{i=1}^{n}x_i^2+\sum_{j=1}^{n}y_j^2} \tag{4}$$

其中，k_x 表示词语 x 的词向量；d_i 和 d_j 分别表示通过式(3)得到的文本 i 和文本 j 的向量；$\mathrm{sim}(d_i,d_j)$ 表示文本 i 和 j 的相似值，相似值越接近 1 表示两篇文本越相似。

⑥ 输出结果。依据文本间相似值和给定阈值 t 判断该文本是否相似，并将结果返回给用户。

3.2 问题描述

文本相似度检测框架中存在以下 2 个方面的问题。

1) 提取的特征词缺乏代表性。在提取特征词阶段，对词位置权重的设置仅按照文本结构简单的设为具有差异的数值，存在较强的主观性且没有合理依据，从而影响词语的总权重，使提取到的特征词不能更准确地表达文本，因此需要设计合理的词位置权重计算方法。

2) 相似度计算结果不够准确。在计算文本相似度阶段，对特征词向量进行叠加或加权平均构建文本向量，将文本相似度计算转化为向量空间相似度度量，这种方法没有考虑到词语之间的语义关联，不能表达文本的深层语义，容易导致计算结果存在偏差，因此需要设计一个融合词语语义关系的相似度计算方法。

4 面向文本的相似度检测方案

针对 3.2 节描述的提取的特征词缺乏代表性、相似度计算结果不够准确这 2 个问题，本文分别提出了基于层次分析法的词语位置加权方法和基于 Pearson 和广义 Dice 系数的相似度计算方法。

4.1 基于层次分析法的词语位置加权方法

层次分析法设置文本各部分对词语重要性影响的权值，通过提高词语位置权重的合理性来提升提取特征词的准确度。AHP 是结合定性和定量分析的综合评估方法，根据决策将问题分解为不同层次的因素，使用定性分析确定元素间的相对重要性，再结合定量分析确定各层次以及各因素的权值，为决策者提供依据，适用于存在主观性和不确定性信息的情况[34-35]。本文利用层次分析法设置文本各部分对词语重要性影响的权值，改进文本相似度检测框架中，特征词提取阶段的词语总权重计算式，通过提高词语位置权重的合理性来提升特征词提取的准确度。该方法的具体步骤如下。

1) 词语位置重要性参数设计

本文设计的相似度检测方案面向的文本类型为学术论文，该类型文本的统一结构包含了论文标题 T、论文摘要 A、论文关键词 K 等，词语位置的重要程度主要由这 3 个因素决定，如式(5)所示。

$$W\mathrm{loc}2=\alpha\mathrm{T}+\beta\mathrm{A}+\gamma\mathrm{K} \tag{5}$$

其中，α、β、γ 为各因素在决定词语位置重要性时所占的比例。

2) 词语位置重要性计算

论文标题通常包含了文章的研究主题、使用方法和应用场景，是论文围绕的核心；论文关键词是作者总结文章重要内容的词语，其对文章的重要性略低于论文标题；论文摘要是从背景、目标、过程、结果对论文的简短概述，包含的词语相对较多，其对文章的重要性相对来说低于论文标题和论文关键词。经分析发现，文本各结构部分对其内容的重要性存在差异，根据 AHP 将论文标题、关键词、摘要作为 3 个因素，计算其成对比较值，即可确定每个因素对文本的重要性。表 1 是由 Saaty 给出的 9 个重要性等级及其量化值，依此构造的成对比较矩阵如表 2 所示。

表 1 9 个重要性等级及其量化值

因素 i 比因素 j 的重要程度	量化值
同等重要	1
稍微重要	3
较强重要	5
强烈重要	7
极端重要	9
两相邻判断的中间值	2、4、6、8

表 2 成对比较矩阵

因素	T	K	A	权重
T	B_{tt}	B_{tk}	B_{ta}	W(T)
K	B_{kt}	B_{kk}	B_{ka}	W(K)
A	B_{at}	B_{ak}	B_{aa}	W(A)

表 2 中，B_{tt} 表示因素 T 与 T 的重要性比值，各因素与其自身的重要性是一样的；B_{tk} 表示因素 T 与 K 的重要性比值，B_{tk} 与 B_{kt} 互为倒数，依次类推，可得到其他两两因素的重要性比值。W(T)、W(K)、W(A)分别表示论文标题、关键词、摘要在决定词语位置重要性时所占的比例，如式(6)所示。

$$W(X)=\sum_{j=t}^{a}\frac{B_{ij}}{\sum_{i=t}^{a}B_{ij}},i\ ,\ j=(\mathrm{t,k,a}),X=(\mathrm{T,K,A}) \tag{6}$$

3) 改进的词语总权重计算

根据式(6)计算得到文本各结构对词语位置的

重要性 $W(\mathrm{T})$、$W(\mathrm{K})$、$W(\mathrm{A})$，将其代入式(5)中可得到式(7)，即得到词语 i 的位置权重，计算式为

$$W\mathrm{loc2}(i)=W(\mathrm{T})\mathrm{T}+W(\mathrm{K})\mathrm{K}+W(\mathrm{A})\mathrm{A} \tag{7}$$

将 $W\mathrm{loc2}(i)$代入原词语总权重计算式(2)中的位置权重 $W\mathrm{loc}(i)$，得到改进后的词语总权重计算式为

$$W2(i)=0.5\mathrm{TF}\text{-}\mathrm{IWF}(i)+0.5W\mathrm{loc2}(i) \tag{8}$$

$W2(i)$作为新的词语总权重用于提取特征词，以在文本相似度检测框架的后续步骤中使用。

4.2　基于 Pearson 和广义 Dice 系数的相似度计算方法

Pearson 相关系数用于衡量 2 个变量之间的线性相关程度，对数据分布比较敏感，适用于正态分布的变量。文献[36]表明语义相似的词向量呈线性关系，且 Word2vec 模型训练的向量更倾向于正态分布。文本相似度检测框架的步骤④采用了 Word2vec 来生成词向量，因此，本文利用 Pearson 相关系数来度量词语间的语义关系，并将其作为广义 Dice 系数的权重改进相似度计算公式。该方法同时考虑了单文本内部和跨文本间的语义关系，提高了文本相似度计算结果的准确性，具体步骤如下。

1) 词语间语义关系度量

特征词提取之后，文本的内容由其特征词代替表示。将特征词输入 Word2vec 模型，每个词被转化为固定维度的向量，每个维度都表示该词语在不同方面的语义信息，例如（$v_1,v_2,\cdots,v_{400}$）。记 k_i 和 k_j 分别为文本 d_i 和 d_j 的特征词，使用 Pearson 相关系数计算词语间语义相似度，如式(9)所示。

$$\mathrm{sim}(k_i,k_j)=\rho(k_i,k_j)=\frac{\mathrm{cov}(I,J)}{\sigma I\sigma J}=\frac{\sum IJ-\dfrac{\sum I\sum J}{N}}{\sqrt{\left(\sum I^2-\dfrac{(\sum I)^2}{N}\right)\left(\sum J^2-\dfrac{(\sum J)^2}{N}\right)}} \tag{9}$$

其中，$\rho\left(k_i,k_j\right)$表示词语 k_i 和 k_j 的相关系数；$\mathrm{cov}(I,J)$ 表示样本协方差；σI 和 σJ 表示样本方差；ρ 的取值范围为[−1, 1]，若相关系数接近 1，两向量之间呈正相关，意味着 2 个词语在语义上越相似，反之，两向量之间呈负相关，意味着 2 个词语在语义上越不相似。

2) 改进的文本相似度计算

该方法中没有将特征词转化为文本向量，而是将式(9)计算的特征词之间的 Pearson 相关系数作为广义 Dice 系数的权重，利用单文本内词语间的不相关性和跨文本间词语的语义相关性，通过两者之间的相对关系得到文本的相似度。由此，改进原始的广义 Dice 系数式(4)，得到式(10)为新的相似度计算式。

$$\mathrm{sim}(d_i,d_j)=\frac{2\sum_{x\in d_i}\sum_{y\in d_j}\rho(x,y)}{\sum\limits_{x_1\in d_i}\sum\limits_{x_2\in d_i}\rho(x_1,x_2)+\sum\limits_{y_1\in d_j}\sum\limits_{y_2\in d_j}\rho(y_1,y_2)} \tag{10}$$

其中，x 和 y 分别表示文本 d_i 和 d_j 的特征词组，$\mathrm{sim}(d_i,d_j)$表示文本 d_i 和 d_j 的相似度。具体的含义为：一组特征词内部两两词语间的相似度越小，该特征词组越能够从多方面充分表达文本内容；同时，两组特征词之间两两词语间的相似度越大，该两组特征词表达的两篇文本内容也越相似。因此，当根据式(10)计算的相似度大于阈值 t 时，表示该两篇文本是相似的。

5　实验与分析

针对本文提出的特征词提取方法和相似度计算方法，分别设计了 2 个对应的实验，来验证本文提出的相似度检测方案的有效性。

5.1　实验 1：特征词提取

1) 实验数据

实验 1 是由复旦大学提供的包含 20 个不同文本类别的中文分类语料。本文从中随机选取已经由人工标注出关键词的农业（agriculture）、艺术（art）、计算机（computer）、经济（economy）、环境（environment）、历史（history）、政治（politics）、航空（space）等 8 类不相关文本各 20 篇以及由这 8 类中每类的两篇文本组成混合文本（mix）16 篇，作为测试数据集。数据集中的每个数据项包括论文标题、摘要、关键词，实验中使用数据集中的关键词字段作为对比项，与实验所提取的特征词相比较来评估各方法的性能。

2) 对比方法及评价指标

实验中选取了经典的 TF-IDF 算法和基于 TF-IDF 改进的 TF-IWF[17] 算法与本文方法（TF-IWF-Location）进行对比。

本文将采用关键词提取领域常用的精确率 P（precision）、召回率 R（recall）、综合指标 F1 值

（F1-score）来评测实验结果，其定义分别如式(11)、～式(13)所示。

$$P=\frac{C_n}{K_n}\times 100\% \tag{11}$$

$$R=\frac{C_n}{B_n}\times 100\% \tag{12}$$

$$\mathrm{F1}=\frac{2PR}{P+R}\times 100\% \tag{13}$$

其中，C_n表示正确提取到的特征词个数，K_n表示提取的所有特征词个数，B_n表示语料中标注的特征词个数。

3) 实验设置

首先提取论文标题、摘要、关键词的内容，并将其合并为一段。然后使用 jieba 分词和哈工大停用词表对合并后的内容分词、去停用词，构建特征词候选词集。

特征词提取是将候选词集中重要度靠前的K个词语输出为特征词。由于数据集中各类文本的长度不一样，且标注的关键词个数不同，为了使实验结果更加客观准确，实验中根据每类文本中标注的关键词个数来调整提取的特征词个数，保证两者之间的差值在 10 之内，通过实验调试，得出每类语料对应所提取合适的特征词个数，如表 3 所示。

表 3　　语料类别与特征词提取个数

语料类别	特征词提取个数/个
agriculture	6
art	8
computer	7
economy	6
environment	7
history	9
politics	7
space	8
mix	8

在本文所提方法 TF-IWF-Location 中引入了层次分析法，将论文标题表示为 T、关键词表示为 K、摘要表示为 A，根据 4.3 节的分析以及表 1 的比例标度，T 比 K 稍微重要，K 比 A 稍微重要，T 比 A 较强重要，通过构造 T、K、A 间的成对比较矩阵得出论文各结构的位置权重参数，如表 4 所示。

4) 实验结果分析

按照以上实验设置进行特征词提取，将不同算法的各项指标以折线图呈现，图 2～图 4 分别是 TF-IDF、TF-IWF、TFIWF-Location 算法的精确率、召回率、F1 值的比较结果。

表 4　　T、K、A 间的成对比较矩阵

因素	T	K	A	权重
T	1	3	5	0.636 2
K	1/3	1	3	0.260 5
A	1/5	1/3	1	0.106 1

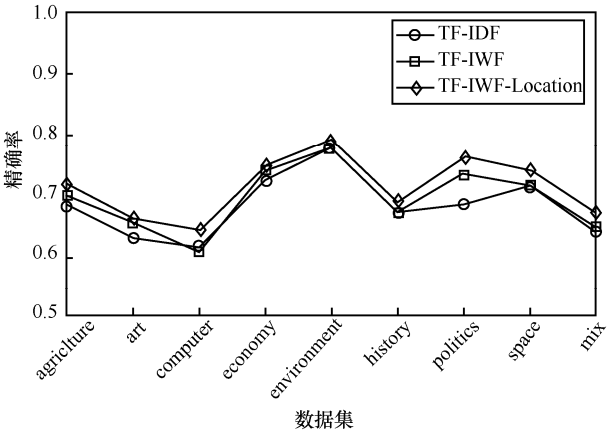

图 2　TF-IDF、TF-IWF、TF-IWF-Location 算法之间的精确率比较

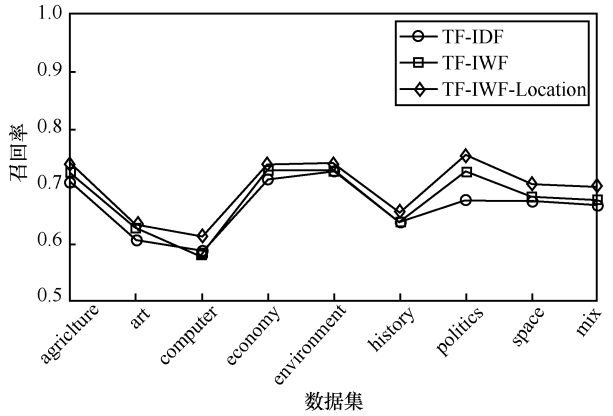

图 3　TF-IDF、TF-IWF、TF-IWF-Location 算法之间的召回率比较

通过图 2～图 4 可知，TF-IWF 算法在 computer 语料上的精确率、召回率和 F1 值略低于 TF-IDF 算法，在 environment 和 history 语料上与 TF-IDF 算法的性能相等，总体上优于 TF-IDF 算法，表明 TF-IWF 算法能够有效地提高提取同类文本集中特征词的准确性。

本文所提方法 TF-IWF-Location 与 TF-IDF、TF-IWF 相比，在精确率、召回率、F1 值等各项指标上均有所提高，特别是在 computer、politics、space、mix 语料上的提高幅度较大，其中，精确率、

召回率、F1 最高分别提高了 7.9%、10.7%、7.8%。结果表明，词语在文章中的结构位置对词语的重要性具有一定的影响，该方法能够较好地提高对学术论文进行特征词提取的准确率。

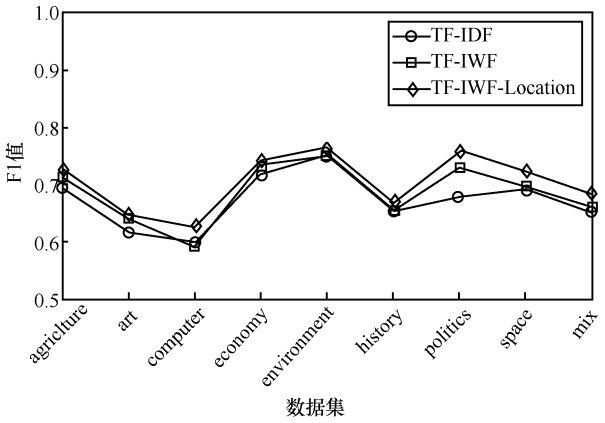

图 4　TF-IDF、TF-IWF、TF-IWF-Location 算法之间的 F1 值比较

5.2　实验 2：文本相似度检测

1) 实验数据

维基百科中文语料库，由中文维基百科中的新闻文章组成，具有质量高、领域广泛且开放的特点。实验中使用的是截至 2021 年 5 月 5 日的中文维基百科语料，大小约 2 GB，包含 392 515 篇文章，以 xml 格式存储。本文以该语料库来训练 Word2vec 模型。

LCQMC 问题语义数据集包含 238 766 对训练文本、8 802 对验证文本和 12 500 对测试文本，这些文本来自百度问答中不同领域的高频相关问题，由人工判定相似的句子对标签为 1，不相似的标签为 0。本文以测试集的 12 500 对句子作为实验的测试数据，通过设置相似度阈值来将计算结果分为相似（1）与不相似（0）两类，与数据集中的标签对比得到实验方法的各项指标对比结果。

2) 对比方法及评价指标

实验中选取了 2 种方法来与本文方法 Pearson-Dice 做对比，一种是传统的基于余弦相似度的方法 Base-Cosine[37]，以特征词叠加后的平均向量表示文本，再计算文本向量之间的余弦相似度；另一种是本文改进之前的方法 Base-Dice，该方法在 Base-Cosine 的基础上，使用广义的 Dice 系数来替代余弦相似度计算文本相似度。

本文将文本相似度检测抽象为相似或不相似的二分类问题。采用二分类领域中常用的准确率（accuracy）和综合指标 F1 值（F1-score）来评估各方法的性能，其定义分别如式(14)和式(15)所示。

$$\mathrm{Acc}=\frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{TN}+\mathrm{FN}}\times 100\% \tag{14}$$

$$\mathrm{F1}=\frac{2PR}{P+R}\times 100\% \tag{15}$$

其中，TP 表示被正确计算为相似句子对的数量，TN 表示被正确计算为不相似的句子对数量，FP 表示被错误计算为相似句子对的数量，FN 表示被错误计算为不相似的句子对数量，P 和 R 分别表示二分类问题中的精确率和召回率，如式(16)和式(17)所示。

$$P=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{16}$$

$$R=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{17}$$

3) 实验设置

下载的维基百科语料为 xml 压缩格式且有较多的不可用数据，不可直接用于训练 Word2vec。先使用 WikiCorpus 方法将文件格式转换为 txt，再通过 Opencc 将文本中的繁体字转为简体字，然后基于正则表达式去除数据中的英文和空格，最后使用 jieba 将分词后的文本输入 Word2vec 模型进行训练。

训练 Word2vec 模型时，有多个参数需要设置。在模式选择上，COWB 模式的速度更快，Skip-gram 模式的效果更好，实验中使用 Skip-gram 模式；滑动窗口的大小为 5，以此构建训练集；最低词频为 5，过滤数据中出现次数低于 5 的词语；词向量维度为 400，官方推荐值为 300～500，此处取中间值；其余参数均为默认。

文本之间的相似性由其相似分布值与阈值的相对大小决定。如果一对不相似文本和一对相似文本的相似值分别为 0.7 和 0.8，那么将相似度阈值设置为 0.75，就能够正确地区分相似和不相似文本。由于数据集和各相似度计算方法会使输出的相似值的分布情况有所差异，因此，实验中分别将相似度阈值设置为 0.50、0.55、0.60、0.65、0.70、0.75、0.80、0.85、0.90 来比较结果。

4) 实验结果分析

按照以上的实验设置进行文本相似度检测，将不同方法的各项指标以折线图展示，图 5 和图 6 分别为 Base-Cosine、Base-Dice、Pearson-Dice 的准确率和 F1 值的比较结果。

由图 5 可知，当相似阈值设置为 0.50、0.55、0.60、0.65、0.70 时，Base-Dice 方法的准确率高于其他 2 种方法；当相似度阈值为 0.75、0.80、0.85、0.90 时，

本文方法的准确率高于其他方法，阈值设置为 0.85 时准确率最高为 75.9%，与 Base-Dice、Base-Cosine 方法相比分别提高了 2.08%和 11.4%。由图 6 可知，仅在相似阈值为 0.9 时，Base-Cosine 方法的 F1 值略高于本文方法的 F1 值，在其余相似度阈值的情况下，均为本文方法 F1 值最高。本文方法在阈值为 0.80 和 0.85 时达到 75.7%和 75.6%，最高值与 Base-Dice、Base-Cosine 方法相比分别提高了 2.8%和 7.1%。

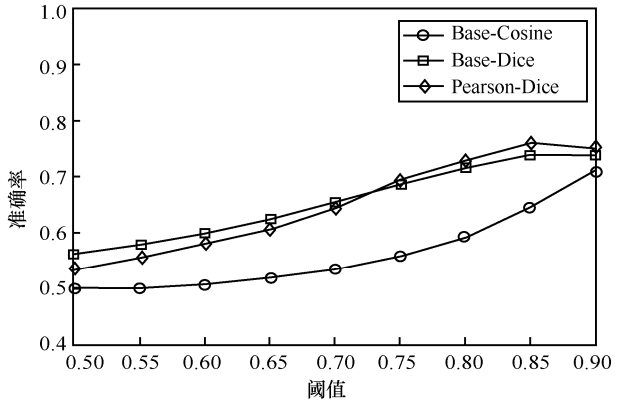

图 5 Base-Cosine、Base-Dice、Pearson-Dice 的准确率比较

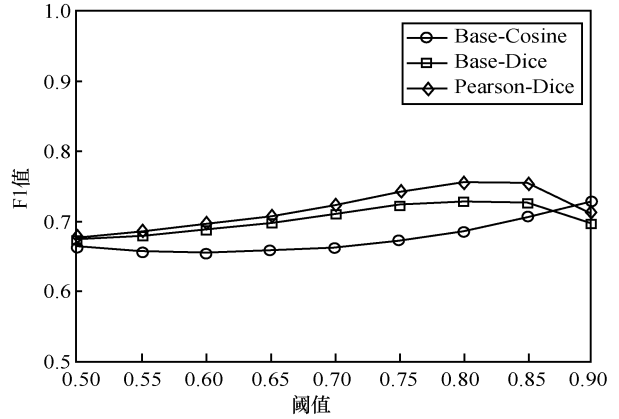

图 6 Base-Cosine、Base-Dice、Pearson-Dice 的 F1 值比较

结果显示，无论是在准确率还是在 F1 值方面，各方法的变化趋势总体上一致，各方法在阈值为 0.80 和 0.85 时对应的准确率、F1 值均达到最高。这表明，该数据集的相似值分布在 0.80～0.85，当阈值设置在这个范围里，能够最好地区分相似或不相似文本，并且均为本文方法性能最优。综上所述，词语间的语义关系在文本相似度计算中发挥了一定的作用，本文所提出的方法是有效可行的。

6 结束语

本文基于向量空间模型的相似度检测算法，在特征词提取阶段提出了基于层次分析法的词语位置加权方法，利用层次分析法确定文本位置对词语的重要性，使提取的特征词更能代表文本；在相似度计算阶段提出了基于 Pearson 和广义 Dice 系数的相似度计算方法，引入了词语语义相似度作为广义 Dice 系数的权重，从而解决了传统方法忽略词语间语义关系的问题。并针对这两点进行改进，分别设计了 2 个对应的实验，与传统方法以及改进前的方法相比，本文提出的方法能够有效地提高计算结果的准确率。下一步将以提高分词准确性继续改进，并进一步探索跨语言的相似度检测，继续提升相似度计算的准确率。

参考文献：

[1] YANG Z X, CHEN Z F, ZHANG P, et al. An information intelligent search method for computer forensics based on text similarity[C]//Proceedings of Proceedings of the 2020 4th International Conference on Cryptography, Security and Privacy. New York: ACM Press, 2020: 79-83.

[2] ALMEIDA C, SANTOS D. Text similarity using word embeddings to classify misinformation[J]. arXiv Preprint, arXiv: 2003.06634, 2020: 63-68.

[3] SEKI K. Cross-lingual text similarity exploiting neural machine translation models[J]. Journal of Information Science, 2021, 47(3): 404-418.

[4] LIANG H Z, LIN K B, ZHU S Z. Short text similarity hybrid algorithm for a Chinese medical intelligent question answering system[C]//Technology-Inspired Smart Learning for Future Education. Singapore: Springer, 2020: 129-142.

[5] PRAKOSO D W, ABDI A, AMRIT C. Short text similarity measurement methods: a review[J]. Soft Computing, 2021, 25(6): 4699-4723.

[6] IRVING R W, FRASER C B. Two algorithms for the longest common subsequence of three (or more) strings[C]//Combinatorial Pattern Matching. Berlin: Springer, 1992: 214-229.

[7] DAMERAU F J. A technique for computer detection and correction of spelling errors[J]. Communications of the ACM, 1964, 7(3): 171-176.

[8] JACCARD P. The distribution of the flora in the alpine zone.1[J]. New Phytologist, 1912, 11(2): 37-50.

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

[10] DEZA M M, DEZA E. Encyclopedia of distances[M]. Berlin: Springer, 2009.

[11] CHANDRASEKARAN D, MAGO V. Evolution of semantic similarity—A survey[J]. ACM Computing Surveys, 2021, 54(2): 1-37.

[12] 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, 1(6): 1-11.
CHEN E J, JIANG E B. Review of studies on text similarity measures[J]. Data Analysis and Knowledge Discovery, 2017, 1(6): 1-11.

[13] 黄文彬, 车尚锟. 计算文本相似度的方法体系与应用分析[J]. 情报理论与实践, 2019, 42(11): 128-134.
HUANG W B, CHE S K. Methodological system and application sce-

narios on text similarity calculation[J]. Information Studies: Theory & Application, 2019, 42(11): 128-134.

[14] LUHN H P. A statistical approach to mechanized encoding and searching of literary information[J]. IBM Journal of Research and Development, 1957, 1(4): 309-317.

[15] BLEI D M, NG A Y, JORDAN M I. Latent dirichlet allocation[J]. Journal of Machine Learning Research, 2001, 3: 601-608.

[16] MIHALCEA R, TARAU P. Textrank: bringing order into text[C]// Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing. [S.n.:s.l.], 2004: 404-411.

[17] 王小林, 杨林, 王东, 等. 改进的 TF-IDF 关键词提取方法[J]. 计算机科学与应用, 2013, 3(1): 64-68.
WANG X L, YANG L, WANG D, et al. Improved TF-IDF keyword extraction algorithm[J]. Computer Science and Application, 2013, 3(1): 64-68.

[18] KIM S W, GIL J M. Research paper classification systems based on TF-IDF and LDA schemes[J]. Human-Centric Computing and Information Sciences, 2019, 9(1): 30.

[19] CHEN W, YU Z T, XIAN Y T, et al. Mining keywords from short text based on LDA-based hierarchical semantic graph model[J]. International Journal of Information Systems in the Service Sector, 2020, 12(2): 76-87.

[20] PUSPANINGRUM E Y, NUGROHO B, SETIAWAN A, et al. Detection of text similarity for indication plagiarism using winnowing algorithm based K-gram and jaccard coefficient[J]. Journal of Physics: Conference Series, 2020, 1569: 022044.

[21] 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 Q L, LI Y M, TANG Q. Similarity computing of documents based on VSM[J]. Application Research of Computers, 2008, 25(11): 3256-3258.

[22] BAO X A, DAI S C, ZHANG N, et al. Large-scale text similarity computing with spark[J]. International Journal of Grid and Distributed Computing, 2016, 9(4): 95-100.

[23] LIU Y, LI D M, DAI C. Short text similarity measure based on double vector space model[J]. International Journal of Database Theory and Application, 2016, 9(10): 33-46.

[24] WANG J Y, XU W H, YAN W H, et al. Text similarity calculation method based on hybrid model of LDA and TF-IDF[C]//Proceedings of the 2019 3rd International Conference on Computer Science and Artificial Intelligence. [S.n.:s.l.], 2019: 1-8.

[25] LIU Y Q, LI Z J. Semantic based text similarity computation[C]// Lecture Notes in Electrical Engineering. Singapore: Springer, 2017: 343-348.

[26] WANG X L, DONG X T, CHEN S X. Text duplicated-checking algorithm implementation based on natural language semantic analysis[C]//Proceedings of 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC). Piscataway: IEEE Press, 2020: 732-735.

[27] 王春柳, 杨永辉, 邓霏, 等. 文本相似度计算方法研究综述[J]. 情报科学, 2019, 37(3): 158-168.
WANG C L, YANG Y H, DENG F, et al. A review of text similarity approaches[J]. Information Science, 2019, 37(3): 158-168.

[28] WANG J P, DONG Y H. Measurement of text similarity: a survey[J]. Information, 2020, 11(9): 421.

[29] SHAHMIRZADI O, LUGOWSKI A, YOUNGE K. Text similarity in vector space models: a comparative study[C]//Proceedings of 2019 18th IEEE International Conference on Machine Learning and Applications (ICMLA). Piscataway: IEEE Press, 2019: 659-666.

[30] 李琳, 李辉. 一种基于概念向量空间的文本相似度计算方法[J]. 数据分析与知识发现, 2018, 2(5): 48-58.
LI L, LI H. Computing text similarity based on concept vector space[J]. Data Analysis and Knowledge Discovery, 2018, 2(5): 48-58.

[31] 陈福, 林闯, 薛超, 等. 短句语义向量计算方法[J]. 通信学报, 2016, 37(2): 11-19.
CHEN F, LIN C, XUE C, et al. Vector semantic computing method study for short sentence[J]. Journal on Communications, 2016, 37(2): 11-19.

[32] MIKOLOV T, CHEN K, CORRADO G, et al. Efficient estimation of word representations in vector space[J]. arXiv Preprint, arXiv:1301.3781, 2013.

[33] 张宇, 刘雨东, 计钊. 向量相似度测度方法[J]. 声学技术, 2009, 28(4): 532-536.
ZHANG Y, LIU Y D, JI Z. Vector similarity measurement method[J]. Technical Acoustics, 2009, 28(4): 532-536.

[34] 邹学强, 包秀国, 黄晓军, 等. 基于层次分析的微博短文本特征计算方法[J]. 通信学报, 2016, 37(12): 50-55.
ZOU X Q, BAO X G, HUANG X J, et al. Calculating the feature method of short text based on analytic hierarchy process[J]. Journal on Communications, 2016, 37(12): 50-55.

[35] 许树柏. 实用决策方法: 层次分析法原理[M]. 天津: 天津大学出版社, 1988.
XU S B. Practical decision-making method: the principle of analytic hierarchy process[M]. Tianjin: Tianjin University Press, 1988.

[36] ZHELEZNIAK V, SAVKOV A, SHEN A, et al. Correlation coefficients and semantic textual similarity[J]. arXiv Preprint, arXiv:1905.07790, 2019.

[37] WESTON J, CHOPRA S, ADAMS K. #TagSpace: semantic embeddings from hashtags[C]//Proceedings of Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). [S.n.:s.l.], 2014: 1822-1827.

[作者简介]

代晓丽（1979− ），女，河南安阳人，北京交通大学博士生，主要研究方向为信息管理。

刘世峰（1970− ），男，河北保定人，博士，北京交通大学教授、博士生导师，主要研究方向为信息管理、大数据分析等。

宫大庆（1982− ），男，山东威海人，博士，北京交通大学副教授，主要研究方向为大数据分析与应用。