文章编号：1003-0077(2022)09-0028-10

融合深层语义和显式特征的中文句子对相似性判别方法

何春辉，胡升泽，张 翀，葛 斌

（国防科技大学 信息系统工程重点实验室，湖南 长沙 410073）

摘 要：中文句子对相似性计算任务旨在利用模型对两个句子的相似性进行判别，在文本挖掘领域有广泛的应用。考虑到现有机器学习方法不能同时兼顾句子对的深层语义特征和显式特征的问题，该文提出融合深层语义和显式特征的中文句子对相似性判别方法。采用 BERT 和全连接网络来获取深层语义向量，再拼接显式特征构造新的特征向量，最后通过分类器完成句子对的相似性判别。实验结果表明，该方法在 3 个公开的中文句子对相似性评测数据集上的性能均优于基线方法。

关键词：语义匹配；特征选取；相似性判别；文本分类；BERT

中图分类号：TP391　　**文献标识码**：A

Chinese Sentence Similarity Measure with Deep Semantics and Explicit Features

HE Chunhui, HU Shengze, ZHANG Chong, GE Bin

(Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha, Hunan 410073, China)

Abstract: The Chinese sentence similarity measure has a wide range of applications in the field of text mining. This paper proposes a Chinese sentence similarity measure by combining deep semantic feotures and explicit features. BERT and feed forward network are used to obtain deep semantic vectors, that are concatenated with explicit features. The final similarity measure is completed through a classifier. The experimental results show that the performance of our proposed method on the three public Chinese datasets is better than all baseline methods.

Keywords: semantic matching; feature selection; similarity discrimination; text classification; BERT

0 引言

日常生活中人们每天接触的新闻内容、社交动态、短信等都是以海量的文本形式呈现的。如何从这些海量的文本数据中挖掘出有价值的信息，已经成为自然语言处理领域的基础研究目标。而文本相似度计算作为自然语言处理的经典任务之一，其在文本聚类、问答匹配、搜索与推荐系统中都发挥着重要的作用。如何从文本中快速挖掘出相似内容，从而攻克去重、抄袭判别、语义匹配等难题，是当前自然语言处理领域中一个热点研究方向。

文本相似度计算的研究范围主要涵盖两个层面：第一个层面是字面相似度计算，即根据两段文本所包含相同字符的多少来判断文本之间是否相似。对于字符相似度的计算方法，目前已经取得了一些成果[1]，主要分为基于字符和基于术语的两类方法。考虑到这些方法是直接对原始文本内容进行字符匹配或通过度量距离来判断文本的相似性，没有考虑词语本身的含义和词语之间的上下文关系，所以不太适合用来计算包含多义词或同义词的句子对之间的相似度。第二个层面是语义相似度计算，即根据两段文本所表达的语义信息是否一致来完成相似性的判别。显然，语义相似度的计算更具挑战性。文本语义相似度计算可以通过对句子的上下文关系以及捕获词语在句中不同含义的信息来进行建模。传统语义相似度计算主要包括基于同义词典[2]和知识库[3]的方法。虽然该类方法可以更充分地考

收稿日期：2021-07-12　　**定稿日期**：2021-08-13

基金项目：国家自然科学基金(61902417)；基础加强基金(2019-JCJQ-JJ-231)

虑文本语义信息来提高相似度计算准确性，但需要依赖人工整理的同义词库和领域知识库，因此实施成本高，且可移植性较差。

句子对的相似性判别属于句间关系的研究范围。它的输入是一对文本，输出类型则与建模方式相关。若采用基于字面相似度或空间向量相似度的方式来建模，一般会输出句子对的相似度值，再结合人工设置的阈值给出相似性判别结果。若采用分类思想来建模，一般会直接输出句子对的分类标签。随着深度学习和预训练技术的发展，采用预训练模型对文本进行向量表征，然后结合深度学习算法[4]来判别句子对的相似性已成为了学术界和工业界的主流方法。受上述方法的启发，为了更好地解决文本相似性的判别难题，本文通过结合特征工程和深度学习算法提出了一种联合深层语义和显式特征的中文句子对相似性判别方法，它属于一种混合方法。新方法既可以捕获文本语义信息，又可以同时兼顾文本的字面特征，从而进一步提升文本相似性的判别准确率。本文的主要贡献如下：

（1）提出了一种既可以捕获深层语义信息，又可以同时兼顾文本字面特征的混合中文句子对相似性判别方法；

（2）采用深度神经网络和特征工程有效融合语义特征和显式特征共同作为分类特征；

（3）新方法通过实验揭示融合 LD/SCC/SC 三个显式特征比融合 SA 的效果会更好。

1　相关研究

文本相似度计算是自然语言处理领域中非常基础的研究方向，主要可分为字面相似度计算、语义相似度计算和混合方法三大类。文本相似度计算研究具有悠久历史，已有很多比较成熟的方法[1,5-6]。相关方法如表 1 所示。

表 1　常用的文本相似度计算方法

一级分类	二级分类	方法名称	说　明
字面相似度计算	基于字符	编辑距离及其变种[7]	两字符串间转换最少操作次数
		最长公共子序列[8]	两字符串最长公共连续子序列
		汉明距离[9]	两个等长字符串在对应位置上不同的数量
		N 元语言模型[10]	两字符串中相同 N 元组数量与总 N 元组数量的比值
		Jaro-Winkler[11]	相同前缀加权的编辑距离变种
	基于术语	Jaccard[12]	两集合中相同词语个数与全部非重复词语个数的比值
		Dice[13]	两集合中相同词语个数的两倍与两个集合中非重复词语个数之和的比值
		Overlap[14]	如果两个集合中有一个是另一个的子集
		匹配系数[15]	两向量相同项都是非零的个数
		余弦相似度[16]	两向量夹角的余弦值
		欧氏距离[17]	向量对应坐标值之间平方差之和的平方根
		曼哈顿距离[18]	两向量对应坐标值的差异之和
		切比雪夫距离[19]	两向量对应坐标值差的绝对值的最大值
		布雷柯蒂斯相异性[20]	用于生物信息学中表征两个群落的差异性
		SimHash[21]	通过对比相似哈希签名计算文档相似度的局部敏感散列算法
		MinHash[22]	根据最小哈希值签名矩阵计算 Jaccard 相似度的局部敏感散列算法
语义相似度计算	基于知识库	ShortestPath[23]	两概念之间最短路径长度倒数
		Hirst&St-Onge[23]	基于词典的词义消岐方法
		Leacock&Chodorow[23]	寻找两个概念之间最短路径，然后在 is-a 层次中找到最大路径
		Wu&Palmer[23]	找两个概念从最近的公共祖先节点到 root 节点的路径长度

续表

一级分类	二级分类	方法名称	说　明
语义相似度计算	基于知识库	Resnik[23]	使用共享父节点信息内容来计算概念词语间的语义相似度
		Lin[23]	描述概念共性所需的信息量与完全描述两个概念所需信息量的比值
		Jiang &Conrath[23]	基于祖先节点和同义词集的信息来表示两个词义的相似程度。
		ESA[24]	显式语义分析是一种计算任意文本之间语义关联的方法
		WLM[25]	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
		SSA[26]	显著语义分析
		TSA[27]	时态语义分析
		CSA[28]	语境语义分析
		NASARI[29]	改进 ESA 方法降维与加权策略
	基于语料库	LSA[30]	潜在语义分析方法
		LDA[31]	一种主题模型,可以将文档集中每篇文档的主题按照概率分布形式给出
		HAL[32]	语义存储模型
		PMIB[33]	基于百度搜索引擎的相似度计算方法
		NGD[34]	一种语义相似度度量,由谷歌搜索引擎对给定的一组关键字返回的命中次数得出
		SH/CODC[35]	基于查询结果片段的方法
		DSSM[36]	微软在 2013 年提出的多塔模型
		CNN 及其变种[37]	采用 CNN 提取特征再结合深层网络和 Softmax 给出标签概率
		LSTM 及其变种[38]	底层采用 LSTM 完成特征提取再结合深层网络计算相似度
		InferSent[39]	Facebook 提出的双塔模型,底层采用 Bi-LSTM 网络结构和 Max 策略效果最好
		Transformer[40]	采用 Transformer 和注意力机制提取特征再结合深层网络或向量空间模型计算相似度
		GenSet[41]	由微软提出的 GenSet,底层使用 GRU 编码器
		BERT-flow[42]	由字节跳动提出的改进版 Sentence-BERT
混合方法	混合	SemSim[43]	采用浅层词法模式抽取捕获词语语义关系
		CODC+PMI[44]	根据词语语义相关性强弱自动选取 CODC 或 PMI 来计算相似度
		MSSA[45]	词义嵌入和消歧方法
		UESTS[46]	基于无监督词对齐的集成语义相似度方法

对字面相似度计算方法来说,不同方法所适应的场景不太一样。有些方法适合用来处理短文本,如 Jaro-Winkler[11]。有些适合处理长文本相似度计算问题,如 SimHash[21]。在语义相似度计算方法中,基于语料库的方法的性能比基于知识库的方法要好,底层采用 Transformer[40]来提取特征的深度学习方法的性能一般会更好。在深度学习中,文本匹配模型采用双塔式网络结构比采用交互式网络结构计算效率要高,但其准确性却不如交互式网络结构。考虑到混合方法同时兼具不同方法的特点,所以泛化性能相对于单一方法来说会更好。受上述工作启发,本文通过联合深层语义信息和显式特征提出了一种新的混合语义相似度计算方法用来解决中文句子对语义相似性判别问题。新方法主要有三个特点:①底层采用 BERT 和全连接的交互式网络结构来提取句子对的深层语义向量。②上层提取一些显式特征,与底层的深层语义向量完成拼接之后再一起输入分类器完成模型训练。③句子对的相似性判别结果由分类器自动给出,无须人工干预。

2　中文句子对相似性判别方法

本文的核心思想是将句子对相似性判别问题转化为文本分类问题。对传统的分类算法而言，通过选取有效的特征，可以提升算法的分类准确率。此外，考虑到语义信息对相似性判别发挥着非常重要作用，本文通过深度神经网络得到低维深层语义向量，与显式特征完成拼接后，共同用来训练文本相似性判别模型。模型整体框架如图 1 所示。

如图 1 所示，整个框架共包含 6 个层次：①输入层，功能是实现句子对的输入；②预处理层，功能是引入空格切分句子中的每一个字，并利用指定的特殊标签[CLS]和[SEP]来完成句子对的拼接；③编码层，功能是采用 BERT 向量编码器对预处理之后的句子进行向量编码，并输出对应的句对向量 $\boldsymbol{X}=[X_1,X_2,X_3,\cdots,X_m]$；④深层语义信息计算层，功能是将编码层输出的句对向量经过全连接网络之后输入 SoftMax 函数，得到二分类标签 Y_1/Y_2 所对应的概率值 P_1/P_2；⑤特征融合层，功能是将人工选取的特征进行归一化后再联合深层语义信息计算层输出的概率，一起作为新的分类特征；⑥分类层，功能主要是利用特征融合层输出的新特征向量作为输入来训练分类器，并给出最终相似性判别结果。

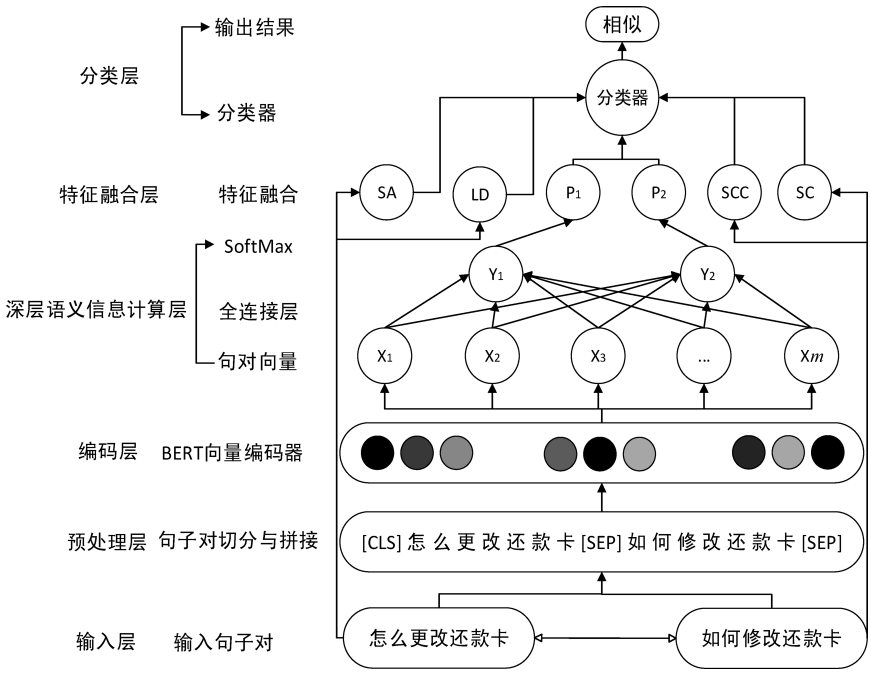

图 1　句子对相似性判别模型的整体架构图

2.1　深层语义向量的计算

当输入层输入一对句子后，经过预处理层就会得到分割后的句子对拼接文本片段。值得注意的是，实验中单句文本的最大长度阈值设置为 50。为了与 BERT 模型的原始训练语料标注模式保持一致，这个文本片段中的字间采用空格分割，句首采用[CLS]分隔符标记，句间采用[SEP]分隔符标记，这样处理的原因是方便直接调用 BERT 预训练模型来完成后续的编码工作，可以把整个深层语义信息计算环节抽象成一个特征降维的过程。在编码层，首先利用 BERT 对预处理之后的句对文本片段做交互式编码转换，得到 $1\times m$ 维的句对向量。BERT 模型采用双向 Transformer 作为编码器实现特征抽取，并结合多头注意力机制捕获更多上下文信息，从而将词语转化为语义特征更丰富的向量形式。自注意力机制输入部分由 Query($\boldsymbol{Q}$)，Key($\boldsymbol{K}$)，Value($\boldsymbol{V}$)三个向量构成，再通过 $\boldsymbol{Q}*\boldsymbol{K}$ 来表示输入部分字向量的相似度，然后通过 D_k 进行合理缩放。最后由 SoftMax 函数做归一化处理得到最终概率分布，进而输出句中所有词向量的权重求和表示。注意力和多头注意力计算过程如式(1)～式(3)所示。

$$\mathrm{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=\mathrm{SoftMax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{D_k}}\right)\boldsymbol{V} \tag{1}$$

$$\mathrm{MutilHead}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=\mathrm{Concat}(\mathrm{head}_1,\cdots,\mathrm{head}_n)\boldsymbol{W}^{O} \tag{2}$$

$$\mathrm{head}_i=\mathrm{Attention}(\boldsymbol{Q}\boldsymbol{W}_i^{Q},\boldsymbol{K}\boldsymbol{W}_i^{K},\boldsymbol{V}\boldsymbol{W}_i^{V}) \tag{3}$$

相关参数矩阵 $\boldsymbol{W}_i^Q \in R^{d_{\text{model}} \times d_k}$，$\boldsymbol{W}_i^K \in R^{d_{\text{model}} \times d_k}$，$\boldsymbol{W}_i^V \in R^{d_{\text{model}} \times d_v}$，$\boldsymbol{W}^O \in R^{hd_v \times d_{\text{model}}}$，实验中，取 $h=12$，$d_k=d_v=d_{\text{model}}/h=64$。经过编码层后得到 $1\times m$ 维的句对向量，需要注意的是，实验中取倒数第二层的输出值作为句向量。然后将其作为输入向量传到全连接网络中实现特征降维。全连接层的相关计算步骤如式(4)、式(5)所示。

$$\boldsymbol{Y}=\begin{bmatrix}Y_1\\Y_2\\\vdots\\Y_k\end{bmatrix}=\begin{bmatrix}W_1^T\\W_2^T\\\vdots\\W_k^T\end{bmatrix}*\begin{bmatrix}X_1\\X_2\\\vdots\\X_m\end{bmatrix}+\begin{bmatrix}b_1\\b_2\\\vdots\\b_k\end{bmatrix} \tag{4}$$

$$Y_j=\boldsymbol{W}_j^T*\boldsymbol{X}_j+b_j \tag{5}$$

实验中，将 1×768 维的句对向量映射成 1×2 维的向量，即取 $m=768$，$K=2$，$\boldsymbol{W}_j^T$ 指第 j 维特征对应的权重系数，b_j 指第 j 维特征对应的偏置。最后将得到的 1×2 维向量输入到 Softmax 函数中输出对应类别的归一化概率。不同类别所对应的概率计算过程如式(6)、式(7)所示。

$$P(Y_i)=\frac{e^{Y_i^L}}{\sum_{j=1}^{2}e^{Y_i^L}} \tag{6}$$

$$0\leqslant P(Y_i)\leqslant 1,\quad \sum_i P(Y_i)=1 \tag{7}$$

因为分类对象为是否相似的二分类问题，所以最终类别数为 2，即 $i=1$ 或者 2。结合式(1)～式(7)可以将任意句子对转化为一个可量化的深层语义向量$[P_1,P_2]$，其中，P_1 指不相似概率，P_2 指相似概率。很多经典方法都是通过比较这两个概率值的大小直接得到最终的判别结果。但受混合方法的启发，本文未直接进行分类，而是将这两个概率值作为深层语义向量，然后再拼接 4 个归一化之后的显式特征来共同构建新的向量 $\boldsymbol{X}'=[f_1,f_2,f_3,f_4]_{1\times4}\oplus[P_1,P_2]_{1\times2}=[f_1,f_2,f_3,f_4,P_1,P_2]_{1\times6}$ 作为模型的输入向量来完成新分类器的训练，从而进一步提升模型的整体性能，其中“⊕”指向量拼接运算。归一化显式特征计算方法见 2.2 节。

2.2 显式特征选取

特征工程是传统机器学习领域中一种重要的建模技术。通过特征工程选取一些有代表性的特征可以提升模型的性能。通过对数据的观察和分析及受文献[47]的启发，最终从句对文本的情感倾向一致性、句对文本长度差比率、句对文本相同字符覆盖率、句对文本同义词覆盖率这 4 个维度选取了比较有代表性的归一化显式特征，并联合深层语义向量来构建分类模型。相关显式特征的详细计算步骤见式(8)～式(11)。情感倾向一致性(SA)计算公式如下：

$$\text{SA}=\begin{cases}1, & \text{如果句子 } A \text{ 和句子 } B \text{ 的情感倾向性相同}\\0, & \text{如果句子 } A \text{ 和句子 } B \text{ 的情感倾向性不同}\end{cases} \tag{8}$$

句子 A 和句子 B 的情感倾向性得分计算见文献[48]。长度差比率(LD)计算如式(9)所示。

$$\text{LD}=\frac{|\text{Len}(A)-\text{Len}(B)|}{\max\{\text{Len}(A),\text{Len}(B)\}} \tag{9}$$

其中，$|\text{Len}(A)-\text{Len}(B)|$指句子 A 与句子 B 长度差的绝对值，$\max\{(\text{Len}(A),\text{Len}(B)\}$指句子 A 与句子 B 长度最大值。相同字符覆盖率(SCC)计算如式(10)所示。

$$\text{SCC}=\frac{T(A,B)}{\min\{\text{Len}(A),\text{Len}(B)\}} \tag{10}$$

其中，$T(A,B)$指句子 A 和句子 B 中相同的字符总数。同义词覆盖率(SC)的计算如式(11)所示。

$$\text{SC}=\frac{\text{句子 } A \text{ 和句子 } B \text{ 中所有名词和动词存在同义关系的词语总数}}{\text{句子 } A \text{ 和句子 } B \text{ 中所有名词和动词的总数}} \tag{11}$$

我们采用开源的同义词词林(扩展版)[49]来判断两个词语之间是否属于同义词。实验中，只选取句子中的名词和动词计算，这样既可以提升计算效率，又可以减少噪声词的干扰。

3 实验设计与结果分析

3.1 数据集与评测指标

3.1.1 数据集

本文为了验证方法的有效性，在 3 个公开中文句子对语义相似度评测数据集上开展了多组对比实验。3 个数据集分别为 LCQMC[50]、BQ-Coupus[51]和 PAWS-X[52]。LCQMC 来源于百度知道领域问题匹配数据集，目的是解决在中文领域大规模问题匹配数据集的缺失问题。BQ-Coupus 是银行金融领域的问题匹配数据集，是目前银行领域问题匹配公开的数据集。PAWS-X 是谷歌发布的包含 7 种语言释义对的数据集，包括 PAWS(英语)与 PAWS-X(多语)。数据集里包含了释义对和非释义对，即识别一对句子是否具有相同的释义。各个数据集的标注方式均一致，即标注出两段文本在语义上是否相似。原始数据集的统计信息如表 2 所示。

表 2　原始数据集的统计信息

数据集名称	训练集大小	验证集大小	测试集大小
LCQMC	238 766	8 802	12 500
BQ-Coupus	100 000	10 000	10 000
PAWS-X	49 401	2 000	2 000

3.1.2　评测指标

因为中文句子对语义相似性的判别属于二分类问题，所以，为了客观地对算法性能进行评估，实验中采用经典的 Precision（P），Recall（R），F_1，Accuracy(Acc)作为评测指标。其中 F_1 值是一个综合指标，F_1 值越大说明算法的综合性能越好，相关指标的计算如式(12)～式(15)所示。

$$P=\frac{\text{被预测为正类的测试样本中真正为正类的样本总数}}{\text{所有被预测为正类的测试样本总数}} \tag{12}$$

$$R=\frac{\text{被预测为正类的测试样本中真正为正类的样本总数}}{\text{所有真正为正类的测试样本总数}} \tag{13}$$

$$F_1=\frac{2PR}{P+R} \tag{14}$$

$$\text{Accuracy}=\frac{\text{正确分类的测试样本总数}}{\text{全部测试样本数量}} \tag{15}$$

此外，为验证不同显式特征对算法耗时的影响，在消融分析中还引入新增单特征耗时指标 t_i，$i\in\{1,2,3,4\}$作为评估标准。其中，t_i 指增加第 i 个显式特征后的算法总耗时。

3.2　实验结果与分析

为充分地验证算法性能，实验中用不同的语义相似度计算模型在 3 个数据集上分别开展了多组对比实验。其中 ESim 和 Linkage[53] 语义相似度计算方法、WMD-char[54]、CNN[54]、Bi-LSTM[54]、DFF-m-char[54]、BiMPM[51]、BERT＋余弦相似度、BERT＋全连接为基线模型。ESim＋显式特征＋分类器、Linkage＋显式特征＋分类器、BERT＋全连接＋显式特征＋分类器为提出的新模型。相关的实验结果分别如表 3～表 5 所示。

表 3　不同模型在 LCQMC 数据集上的实验结果

模型名称	P	R	F_1	Acc
ESim	0.750	0.750	0.750	0.754
ESim＋显式特征＋AdaBoost	**0.790**	**0.780**	**0.785**	**0.778**
Linkage	0.810	0.810	0.810	0.808
Linkage＋显式特征＋决策树	**0.870**	**0.880**	**0.875**	**0.876**
WMD-char [54]	0.640	0.790	0.708	0.600
CNN [54]	0.670	0.910	0.775	0.735
Bi-LSTM [54]	0.780	0.940	0.850	0.834
DFF-m-char [54]	0.770	**0.950**	0.848	0.831
BERT＋余弦相似度	0.850	0.860	0.855	0.856
BERT＋全连接	**0.890**	0.880	0.885	0.890
BERT＋全连接＋显式特征＋贝叶斯	**0.890**	0.890	**0.890**	**0.893**

续表

表 4　不同模型在 BQ-Coupus 数据集上的实验结果

模型名称	P	R	F_1	Acc
ESim	0.710	0.680	0.695	0.679
ESim＋显式特征＋随机森林	**0.720**	**0.720**	**0.720**	**0.720**
Linkage	0.700	0.690	0.695	0.683
Linkage＋显式特征＋决策树	**0.710**	**0.710**	**0.710**	**0.708**
CNN [51]	0.680	0.710	0.690	0.685
Bi-LSTM [51]	0.750	0.705	0.727	0.735
BiMPM [51]	0.820	0.811	0.817	0.818
BERT＋余弦相似度	0.818	0.819	0.818	0.829
BERT＋全连接	0.830	0.830	0.830	0.839
BERT＋全连接＋显式特征＋随机森林	**0.840**	**0.840**	**0.840**	**0.843**

表 5　不同模型在 PAWS-X 数据集上的实验结果

模型名称	P	R	F_1	Acc
ESim	0.530	0.530	0.530	0.554
ESim＋显式特征＋决策树	**0.660**	**0.590**	**0.623**	**0.638**
Linkage	0.540	0.530	0.535	0.562
Linkage＋显式特征＋随机森林	**0.650**	**0.590**	**0.618**	**0.634**
BERT＋余弦相似度	0.700	0.650	0.674	0.694
BERT＋全连接	0.750	0.750	0.750	0.759
BERT＋全连接＋显式特征＋随机森林	**0.760**	**0.760**	**0.760**	**0.766**

综合表 3、表 4 和表 5 的结果可知，对所有基线方法而言，在 3 个数据集上不同算法的性能差距较大，BERT＋余弦相似度方法的性能相对来说稍好一点，但是综合性能最佳的是 BERT＋全连接方法。结果表明，所有新提出的模型，通过融合显式特征并加上不同的分类器，可以在原始基线模型的基础上

进一步提升性能。在LCQMC数据集上性能最好的是BERT＋全连接＋显式特征＋贝叶斯方法，它的平均准确率达到了89.3%。在BQ-Coupus和PAWS-X数据集上性能最好的是BERT＋全连接＋显式特征＋随机森林方法，它们的平均准确率分别达到了84.3%和76.6%。通过对比分析不同数据集上的实验结果发现，对于ESim和Linkage基线模型而言，新模型在融合显式特征之后的平均Accuracy增幅可以达到2.4%～8.35%。但新模型对BERT＋全连接的基线模型而言，在显著性水平$\alpha=0.05$的情况下，实验结果表明，新方法在融合相关显式特征之后的平均Accuracy只是小幅提升了0.3%～0.7%，整体差异性并不大。出现这种现象的主要因素可能与选取的外部显式特征不够强大有一定的关系，或许未来通过进一步优化外部显式特征，可以大幅度提升新模型的整体性能。次要因素有可能是BERT＋全连接模型的自身性能较强大所导致，因为预训练向量中或许会提取部分显式特征信息。但ESim和Linkage方法通过融合外部显式特征来提升性能是一种非常有效的途径，在不同的数据集上，准确率提升都比较明显。此外，实验结果还表明，最佳分类器的选择会与数据分布有关，因为不同类型的数据，内容结构和长短存在一定的差异性，因此对应的最佳分类器也会有所不同。实验中所采用的BERT＋全连接模型核心超参数设置如表6所示。

表6 BERT＋全连接模型核心超参数设置

参数名称	参数取值	参数名称	参数取值
train_epochs	10	batch_size	128
learning_rate	0.000 05	layer _ indexes	[－2]
max_seq_len	50	pretrained _model	Chinese_L-12_H-768_A-12

3.3 消融分析

考虑到数据规模和数据质量的影响，最终选择在LCQMC数据集上完成消融分析实验。消融分析实验中采用平均Accuracy和新增单特征耗时指标作为评估标准。在LCQMC数据集上采用不同基线模型与融合显式特征之后的模型所对应的实验结果如图2所示。

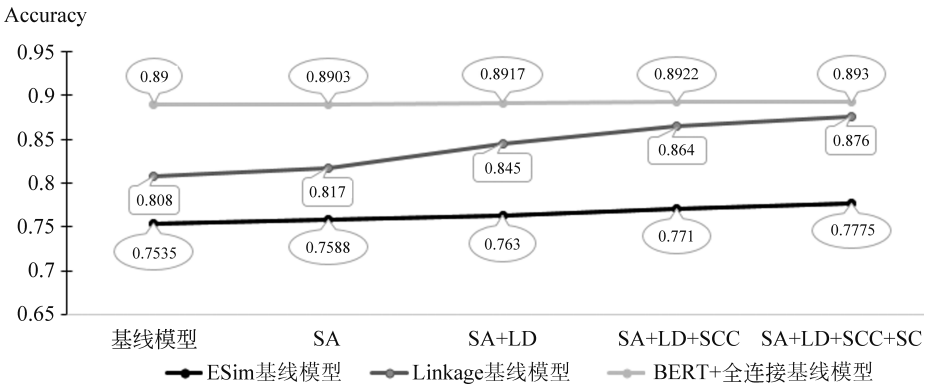

图2 不同基线模型与融合显式特征之后的实验结果

根据图2的实验结果可知，新模型在融合不同的外部显式特征之后可以在不同程度上提升原始基线模型的性能。尤其是在融合LD/SCC/SC三个显式特征之后，模型的整体性能提升比较明显，但是融合SA特征的性能提升幅度并不明显。根据图2中的结果可以看出，融合外部显式特征之后，Linkage基线模型的平均准确率提升比较明显，但是对BERT＋全连接模型的性能提升较小。出现这种现象的可能原因是Linkage算法相对特征提取能力更强的BERT＋全连接算法而言，在特征提取方面本身就存在一定的差距。因此，通过融合外部显式特征之后，可以大大地改善Linkage算法特征提取方面的不足，从而提升模型的性能。此外，为了从计算耗时方面揭示模型的性能，还采用了新增单特征耗时指标来对模型性能进行评估，主要以BERT＋全连接＋显式特征＋贝叶斯模型作为基准模型来展开实验，相关实验结果如图3所示。

由图3实验结果可知，新模型在融合不同的显式特征之后相对原始基线模型而言都会产生额外的耗时。其中，融合SA特征所产生的额外耗时非常高，因为需要计算句子对之间的情感倾向一致性指

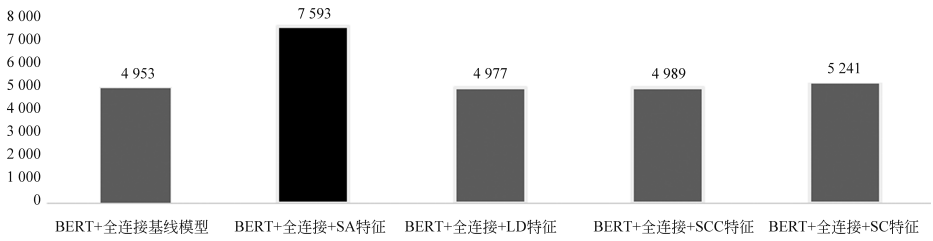

图 3　新增单特征 BERT＋全连接模型的耗时情况(单位：s)

标，它会涉及一些诸如分词的复杂步骤。而融合 LD/SCC/SC 显式特征所产生的额外耗时较少。因此，综合图 2 和图 3 的实验结果来看，虽然 SA 特征能在一定程度上提升模型的性能，但它会大大增加模型的整体耗时。如果对时效性要求较高，建议只融合 LD/SCC/SC 三个显式特征，因为它们对模型的性能提升明显且新增耗时较少。此外，考虑到 LCQMC 是来源于百度知道领域问题匹配数据集，它所涵盖的范围比较广泛，且问题内容一般都是偏客观，只有少部分问题会带有情感倾向性，这可能也是导致融合 SA 显式特征性能提升不明显的原因之一。相对而言，LD/SCC/SC 三个显式特征因为跟数据集所属领域关系不大，且可以直接根据问题字面内容提取有用的信息，因此它们对模型的性能提升比较明显。

3.4　案例分析

为了更加直观地反映算法的性能，从 LCQMC 数据集中选取了 3 组具有代表性的句子对来做案例分析。整个案例分析主要围绕 ESim 算法和新方法(BERT＋全连接以及 BERT＋全连接＋显式特征＋分类器)和人工标注结果来展开，相关结果如表 7 所示。其中，0 代表两个句子含义不相似，1 代表两个句子含义相似。

表 7　3 组样例的对比分析结果

样例句子对内容	ESim 算法判别结果	BERT＋全连接判别结果	新方法判别结果	人工标注结果
(1) 开初婚未育证明怎么弄 (2) 初婚未育证明怎么开	1	1	1	1
(1) 手机微信内容可以同步到电脑上吗 (2) 电脑微信和手机微信可以同步吗	0	1	1	1
(1) 犯太岁是什么意思 (2) 害太岁是什么意思	0	1	0	0

根据表 7 的案例分析结果可知，类似于例案(1)，这种比较常规的样本，三种方法都可以准确地判别结果，但是对于稍微复杂或者带有歧义的案例(2)和案例(3)，ESim 算法和 BERT＋全连接的方法就很有可能会出现误判。值得庆幸的是，得益于外部显式特征的辅助，BERT＋全连接＋显式特征＋分类器的新方法却可以准确地给出判别结果，这充分说明了新方法的有效性。

4　结论与展望

本文提出了一种联合深层语义信息和显式特征的中文句子对相似性判别法。综合考虑了句子对相似度判别问题中深层语义信息、情感倾向一致性及句子对的长度差等显式特征的影响。在 3 个公开中文数据集上的实验结果表明，通过融合显式特征的新模型性能要优于原始基线模型，充分验证了本方法的有效性。

考虑到当前实验数据集和所用 BERT 预训练模型都是针对中文语料且选择的显式特征也是按照中文特性来进行计算的，后续工作中，可以研究其他类型文本或混合文本的相似性判别方法。此外，根据不同的数据集去构建更优的显式特征进一步提升模型性能，也是后续研究中的一个重要方向。

参考文献

[1] 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(03)：158-168.
[2] 杨泉，孙玉泉.基于《同义词词林》深度的词义相似度计算研究[J].计算机工程与应用，2020，56(17)：48-54.
[3] 施凯伦. 知识库与语料库相结合的语义相似度的研究与实现[D].北京：北京交通大学硕士学位论文，2016.
[4] 杨晨. 基于神经网络的短文本语义相似度计算方法研究[D].成都：电子科技大学硕士学位论文，2020.
[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，2017，1(06)：1-11.
[6] Chandrasekaran D，Mago V. Evolution of semantic similarity：A survey[J]. ACM Computing Surveys (CSUR)，2021，54(2)：1-37.

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

[8] Melamed I D. Automatic evaluation and uniform filter cascades for inducing *n*-best translation lexicons[J]. arXiv preprint cmp-lg/9505044, 1995.

[9] 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.

[10] Kondrak G. *N*-gram similarity and distance[C]// Proceedings of the International Symposium on String Processing and Information Retrieval. Springer, Berlin, Heidelberg, 2005: 115-126.

[11] Hu S, He C, Zhang C, et al. Incremental rapidly grouping aggregation method for similar web news headline[J]. Journal of Physics: Conference Series. IOP publishing, 2020, 1453(1): 012153, DOI:10.1088/1742-6596/1453/1/012153.

[12] Bag S, Kumar S K, Tiwari M K. An efficient recommendation generation using relevant Jaccard similarity [J]. Information Sciences, 2019, 483: 53-64.

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

[14] Lawlor L R. Overlap, similarity, and competition coefficients[J]. Ecology, 1980, 61(2): 245-251.

[15] Heltshe J F. Jackknife estimate of the matching coefficient of similarity[J]. Biometrics, 1988: 447-460.

[16] Rahutomo F, Kitasuka T, Aritsugi M. Semantic cosine similarity[C]//Proceedings of the 7th International Student Conference on Advanced Science and Technology ICAST, 2012.

[17] Elmore K L, Richman M B. Euclidean distance as a similarity metric for principal component analysis [J]. Monthly Weather Review, 2001, 129(3): 540-549.

[18] Krause E F. Taxicab geometry[J]. The Mathematics teacher, 1973, 66(8): 695-706.

[19] Mousa A, Yusof Y. An improved Chebyshev distance metric for clustering medical images[C]//Proceedings of the AIP Conference Proceedings. AIP Publishing LLC, 2015, 1691(1): 040020.

[20] Bray J R, Curtis J T. An ordination of the upland forest communities of southern Wisconsin[J]. Ecological Monographs, 1957, 27(4): 325-349.

[21] Sadowski C, Levin G. Simhash: Hash-based similarity detection[J]. Technical Report, Google, 2007: 1-10.

[22] Shrivastava A, Li P. Indefense of minhash over simhash[C]//Proceedings of the Artificial Intelligence and Statistics. PMLR, 2014: 886-894.

[23] Gomaa W H, Fahmy A A. A survey of text similarity approaches[J]. International Journal of Computer Applications, 2013, 68(13): 13-18.

[24] Gabrilovich E, Markovitch S. Computing semantic relatedness using Wikipedia-based explicit semantic analysis[C]//Proceedings of the IJCAI. 2007, 7: 1606-1611.

[25] Witten I H, Milne D N. An effective, low-cost measure of semantic relatedness obtained from Wikipedia links[C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2008: 25-30.

[26] Hassan S, Mihalcea R. Semantic relatedness using salient semantic analysis[C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2011, 25(1): 884-889.

[27] 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, 2011: 337-346.

[28] Benedetti F, Beneventano D, Bergamaschi S. Context semantic analysis: A knowledge-based technique for computing inter-document similarity[C]//Proceedings of the International Conference on Similarity Search and Applications. Springer, Cham, 2016: 164-178.

[29] Camacho-collados J, Pilehvar M T, Navigli R. Nasari: a novel approach to a semantically: aware representation of items[C]//Proceedings of the Conference of the NAACL: Human Language Technologies, 2015: 567-577.

[30] 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.

[31] Bin G E, He C, Hu S, et al. Chinese news hot subtopic discovery and recommendation method based on key phrase and the LDA model[J].DEStech Transactions on Engineering and Technology Research, 2018: 349-358.

[32] Lund K. Semantic and associative priming in high-dimensional semantic space[C]//Proceedings of the 17th Annual Conferences of the Cognitive Science Society, 1995: 660-665.

[33] 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):1056-1060.

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

[35] Chen H H, Lin M S, Wei Y C. Novel association measures using web search with double checking [C]//Proceedings of the ACL, 2006: 1009-1016.

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

[37] Li M, Clinton G, Miao Y, et al. Short text classification via knowledge powered attention with similarity

matrix based CNN[J]. arXiv preprint arXiv:2002.03350, 2020.

[38] Neculoiu P, Versteegh M, Rotaru M. Learning text similarity with Siamese recurrent networks[C]//Proceedings of the 1st Workshop on Representation Learning for NLP, 2016: 148-157.

[39] Conneau A, Kiela D, Schwenk H, et al. Supervised learning of universal sentence representations from natural language inference data[J]. arXiv preprint arXiv:1705.02364, 2017.

[40] 赵梦凡. 基于 Transformer 的文本语义相似度算法研究[D].湘潭：湘潭大学硕士学位论文,2020.

[41] Subramanian S,Trischler A, Bengio Y, et al. Learning general purpose distributed sentence representations via large scale multi-task learning[J]. arXiv preprint arXiv:1804.00079, 2018.

[42] Li B, Zhou H, He J, et al. On the sentence embeddings from pre-trained language models[J]. arXiv preprint arXiv:2011.05864, 2020.

[43] Bollegala D, Matsuo Y, Ishizuka M. Measuring semantic similarity between words using web search engines[J]. WWW, 2007, 7(2007): 757-766.

[44] 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.

[45] Ruas T, Grosky W, Aizawa A. Multi-sense embeddings through a word sense disambiguation process[J]. Expert systems with applications, 2019, 136: 288-303.

[46] Hassan B, Abdelrahman S E,Bahgat R, et al. Uests: An unsupervised ensemble semantic textual similarity method[J]. IEEE Access, 2019, 7: 85462-85482.

[47] Jiang Z, Zhu T, Man L. ECNU: One stone two birds: ensemble of heterogenous measures for semantic relatedness and textual entailment[C]//Proceedings of the International Workshop on Semantic Evaluation, 2014: 271-277.

[48] Ge B, He C H, Zhang C, et al. Classificationalgorithm of Chinese sentiment orientation based on dictionary and LSTM[C]//Proceedings of the 2nd International Conference on Big Data Research. Association for Computing Machinery, New York, NY, USA, 2018, 119-126.

[49] 哈工大社会计算与信息检索研究中心.同义词词林(扩展版)[EB/OL]. https://www.ltp-cloud.com/download.[2021-06-09].

[50] Liu X, Chen Q, Deng C, et al.Lcqmc: A large-scale Chinese question matching corpus[C]//Proceedings of the 27th International Conference on Computational Linguistics, 2018: 1952-1962.

[51] Chen J, Chen Q, Liu X, et al. The BQ corpus: A large-scale domain-specific Chinese corpus for sentence semantic equivalence identification[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2018: 4946-4951.

[52] Yang Y, Zhang Y, Tar C, et al. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification[J]. arXiv preprint arXiv:1908.11828, 2019.

[53] ZHang H, Liu L, Jiang H, et al. TexSmart: A text understanding system for fine-grained NER and enhanced semantic analysis[J].arXiv preprint arXiv: 2012.15639, 2020.

[54] Zhang X, Lu W, Li F, et al. Deep feature fusion model for sentence semantic matching[J]. Computers, Materials and Continua, 2019, 61(2): 601-616.

何春辉(1991—)，硕士，算法工程师，主要研究领域为自然语言处理和机器学习算法应用。
ORCID: http://orcid.org/0000-0003-1505-1620
E-mail: xtuhch@163.com

张翀(1982—)，博士，副研究员，主要研究领域为大数据分析与情报智能。
E-mail: chongzhang@nudt.edu.cn

胡升泽(1981—)，通信作者，博士，研究员，主要研究领域为大数据分析与社会计算。
E-mail: springsun.hu@gmail.com