计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2024,33(7):103−111 [doi: 10.15888/j.cnki.csa.009538]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于 SimCSE 框架融合预训练模型层级特征的文本匹配①

盛成城[1], 陈进东[2,3], 张　健[2,3]

[1](北京信息科技大学 计算机学院, 北京 100192)
[2](北京信息科技大学 经济管理学院, 北京 100192)
[3](智能决策与大数据应用北京市国际科研合作基地, 北京 100192)
通信作者: 陈进东, E-mail: j.chen@bistu.edu.cn

摘　要: SimCSE 框架仅使用分类令牌[CLS]token 作为文本向量, 同时忽略基座模型内层级信息, 导致对基座模型输出语义特征提取不充分. 本文基于 SimCSE 框架提出一种融合预训练模型层级特征方法 SimCSE-HFF (SimCSE with hierarchical feature fusion, SimCSE-HFF). SimCSE-HFF 基于双路并行网络, 使用短路径和长路径强化特征学习, 短路径使用卷积神经网络学习文本局部特征并进行降维, 长路径使用双向门控循环神经网络学习深度语义信息, 同时在长路径中利用自编码器融合基座模型内部其他层特征, 解决模型对输出特征提取不充分的问题. 在 STS-B 的中文与英文数据集上, SimCSE-HFF 方法效果在语义相似度 *Spearman* 和 *Pearson* 相关性指标上优于传统方法, 在不同预训练模型上均得到提升; 在下游任务检索问答上也优于 SimCSE 框架, 具有更优秀的通用性.

关键词: 文本匹配; SimCSE; 特征融合; 自编码器; 并行网络

引用格式: 盛成城,陈进东,张健.基于 SimCSE 框架融合预训练模型层级特征的文本匹配.计算机系统应用,2024,33(7):103–111. http://www.c-s-a.org.cn/1003-3254/9538.html

Text Matching Based on SimCSE Framework Fused with Pre-trained Model Internal Hierarchical Features

SHENG Cheng-Cheng[1], CHEN Jin-Dong[2,3], ZHANG Jian[2,3]

[1](Computer School, Beijing Information Science and Technology University, Beijing 100192, China)
[2](School of Economics and Management, Beijing Information Science and Technology University, Beijing 100192, China)
[3](Beijing International Science and Technology Cooperation Base for Intelligent Decision and Big Data Application, Beijing 100192, China)

Abstract: The simple contrastive learning of sentence embedding (SimCSE) framework only uses the classification [CLS]tokens as text vectors, and it also neglects the hierarchical information within the base model, which results in insufficient extraction of semantic features from the base model output. Based on the SimCSE framework, this study proposes a method that fuses hierarchical features of pre-trained models, SimCSE with hierarchical feature fusion (SimCSE-HFF). SimCSE-HFF is based on a dual-path parallel network, using short and long paths to strengthen feature learning. The short path uses a convolutional neural network to learn local text features and perform dimensionality reduction, while the long path uses a bidirectional gated recurrent neural network to learn deep semantic information. Additionally, in the long path, an autoencoder is used to fuse features from other layers within the base model, solving the problem of insufficient extraction of output features by the model. On the Chinese and English datasets of spring tools suite-bundle (STS-B), the SimCSE-HFF method outperforms traditional methods in terms of semantic similarity

① 基金项目: 国家重点研发计划 (2019YFB1405303); 北京市属高等学校优秀青年人才培育计划 (BPHR202203233); 国家自然科学基金面上项目 (72174018)
收稿时间: 2024-01-10; 修改时间: 2024-02-07; 采用时间: 2024-02-23; csa 在线出版时间: 2024-06-05
CNKI 网络首发时间: 2024-06-07

Spearman and *Pearson* correlation metrics, showing improvements on different pre-trained models. Additionally, it also outperforms the SimCSE framework in downstream task retrieval-based question answering, demonstrating better versatility.

Key words: text matching; SimCSE; feature fusion; autoencoder; parallel network

1 引言

文本匹配是自然语言处理领域中一项重要的任务, 其在问答系统等领域中扮演着关键角色[1]. 在问答系统中, 文本匹配技术通过相似度计算, 在海量的知识库或文本数据中匹配相应的答案或解决方案. 影响文本匹配效果的关键因素是文本向量的质量. 早期的研究主要是基于词频统计的方法, 如 TF-IDF、VSM、BM25 等, 但统计方法通常忽略了上下文和语义信息. 随着人工智能的不断发展, 基于深度学习的文本向量生成模型发展迅速, 如 Word2Vec、ELMO[2]等基于神经网络的模型都有不错的效果. 另外, 也有越来越多的工作是基于预训练模型来实现. 预训练模型通过大量的语料学习, 对上下文建模, 模型学习到丰富的语言知识和语义表示, 能够更好地理解上下文关系, 如 BERT[3]等预训练模型可以生成高质量的文本向量用于下游任务.

预训练模型在进行下游任务时需要微调, 单纯地使用预训练模型可能因为各向异性[4]等问题导致文本向量质量下降, 因此不少研究也提出了针对文本向量生成的微调框架. SimCSE[5]是一种无监督对比学习框架, 拥有良好的文本向量生成能力. 该框架使用预训练模型作为基座模型, 使用 Dropout 方式对同一样例进行两次处理作为正例, batch 内其他样例作为负例, 在向量空间中拉近正例, 推远负例. 但 SimCSE 框架有两个问题.

(1) SimCSE 框架对基座模型输出语义特征提取不充分, 只将[CLS]token 取出并做池化处理. [CLS]token 所对应的嵌入作为文本向量, 因为它自身初始不带信息, 同时汇集了全句信息, 更加公平地融合了文中的语义. 但是如 ColBERT[6]等认为仅使用[CLS]作为文本向量代表是不充分的, 在如 ColBERT、Poly-encoders[7]等方法中, 都在预训练模型输出后做了进一步的特征学习. 如 ColBERT 在采用 query 和 doc 的多个向量相似度累计得分, 而 Poly-encoders 则对于输入的 query 添加了可学习的向量, 使用注意力机制进一步学习, 提高模型效果及推理速度.

(2) SimCSE 框架忽略基座模型内层级信息. 对于深度学习网络, 经验上人们认为深度神经网络的浅层网络与深层网络包含不同的特征, 基于 Transformer[8]的预训练模型通常具有多层结构[9,10]. Jawahar 等[11]对 BERT 的 12 层输出做了不同的实验, 以确定不同层数具体学习到各类特征, 认为浅层学习到了短语级信息, 而中层学习了更多的语法特征, 深层则更具语义特征. 无监督方法 SimCSE 只使用基座模型最后层输出, 忽略模型内部信息, 导致效果不佳.

综上所述, 本文选择 SimCSE 架构的无监督对比学习模型, 提出一种融合预训练模型层级特征的方法 SimCSE-HFF, 使用短路径和长路径双路并行网络强化对特征的学习. 短路径使用卷积神经网络学习文本局部特征并进行降维; 长路径使用双向门控循环神经网络学习深度语义信息, 解决模型对输出特征提取不充分的问题. 另外, 在长路径中利用自编码器融合模型内部其他层特征, 获取更丰富的特征信息. 通过双路并行网络的特征学习, SimCSE-HFF 框架能够优化文本向量生成, 更好地判断文本语义相似度, 同时提升下游任务能力.

2 相关工作

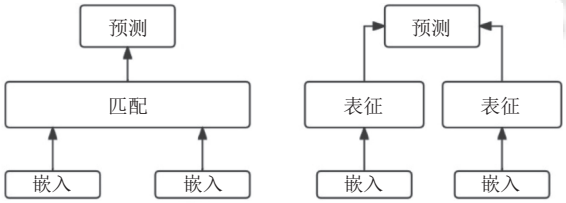
传统的文本匹配方法在上下文理解和长文本处理等方面存在缺陷, 随着深度学习的发展, 基于深度学习的文本匹配方法占据了主导. 基于深度学习的文本匹配方法分为两种: 一种是交互型文本匹配, 另一种是表征型文本匹配.

2.1 交互型文本匹配

如图 1(a) 所示, 交互型模型的特点是使用网络将对应的两段文本, 进行交互、对比, 得到各文本强化后的向量表征, 或者直接得到统一的向量表征, 之后预测二者关系. 基于交互如 ESIM[12], 使用 LSTM[13]和句子间软注意力获取文本特征, 由于 LSTM 网络通常包含大量的参数, 在处理大规模数据集时, 训练和推理的计算代价较高. MatchPyramid[14]则使用 CNN 作为文本的特征提取器, CNN 仅能使用在滤波器尺寸范围内的局

部信息进行特征提取, 导致对于一些依赖于全局语义理解的任务表现不佳.

基于预训练模型的交互型文本匹配方法, 由于其自身模型巨大的参数以及对海量预料的学习, 模型拥有强大的特征提取能力. 大量的研究都是基于类似 Transformer, BERT 等预训练模型上的工作. 李广等使用 BERT 模型获取句子特征并设计其他交互结构获取更丰富的语义匹配信息[15]; 后琦等则融合了关键词、意图等方面的匹配信息[16]. BERT 不仅可以做基于表示模型的特征提取, 也可以直接做基于交互的模型, 其将正负例用分隔符隔开并作为输入, 并对输出进行二分类, 判断二者是否相似.



(a) 交互型 (b) 表征型

图 1 交互型模型与表征型模型架构图

值得注意的是, 基于交互的模型注重结构上对文本特征进行提取与交互, 使得其在同参数下效果更优, 但由于对于每一对正负例都需要输入模型计算, 通常需要更多的时间与计算成本.

2.2 表征型文本匹配

如图 1(b) 所示, 基于表征的文本匹配, 在初始阶段对两个文本各自单独处理, 通过深层的神经网络进行编码, 得到文本的表征, 再对两个表征进行匹配计算得到两个文本的相似度.

比较出色的模型有 DSSM[17], 采用 MLP 作为特征提取器, 也可以将其替换为其他类型网络, 可以适应不同的语义匹配任务, 具有很高的可扩展性和通用性. CDSSM[18]作为 DSSM 的改进型, 采用 CNN 替换了 MLP 来缓解丢失上下文关系的问题. MV-LSTM[19]则使用了 LSTM 网络进行特征提取, 并在匹配阶段对相似度矩阵进行池化操作, 最终分类得到结果.

基于预训练的表征型文本匹配方法, 比较出色的是 SBERT[20]. 采用双塔构型, SBERT 使用 BERT 预训练模型进行特征提取, 获得文本嵌入后经过池化层, 最后计算余弦相似度. SBERT 利用多种特征融合技术, 包括平均池化、最大池化和 CLS 标记, 将多个句子级别的表示融合成一个固定长度的句向量, 可以更好地表示整个句子的语义. SimCSE[3]采用无监督对比学习的方式进行训练. SimCSE 使用了 Dropout 的方式来构建正例, 将一个样本经过 encoder 两次, 通过 Dropout 对不同神经元失活来获得两个不同但相似的句向量, 得到了一个正例对, 负例则是同一个 batch 里的其他句子. 在投影空间内推远负例拉近正例来学习特征.

不过相较于交互型文本匹配方法, 表征型文本匹配方法如 SimCSE 都没有对模型输出特征作进一步学习, 对基座模型输出语义特征提取不充分, 只将[CLS] token 取出并做池化处理. 另外, SimCSE 对基座模型内层级信息利用不充分, 导致本应有用的特征被浪费掉, 制约模型性能. 在 Kim 等[21]的工作里, 希望在不引入外部资源或显示的数据增强的情况下, 利用 BERT 内部信息进行对比, 以提高句子表示的质量. 通过使用两个 BERT, 将一个固定参数 BERT 的各层结果均匀采样和一个微调参数 BERT 的[CLS]输出构造成正例. Jawahar 等[11]对 BERT 的 12 层输出做了不同的实验, 以确定不同层数具体学习的特征, 研究发现浅层学习到了短语级信息, 而中层学习了更多的语法特征, 深层则更具语义特征. 这也引起了本文对模型自身信息的重视, 本文希望增加对模型自身信息的利用, 融合模型层级特征, 提升文本向量生成质量.

3 模型方法设计

3.1 模型总体设计

SimCSE 框架只使用[CLS]token 作为文本向量代表, 同时忽略了基座模型内层级信息, 导致对基座模型输出语义特征提取不充分, 限制了模型的能力. 因此, 本文提出一种融合预训练模型层级特征方法 SimCSE-HFF, 分为 3 层, 嵌入层, 特征融合层, 更新层, 如图 2 所示.

嵌入层由预训练模型组成, 特征融合层由并行双路网络组成, 更新层由 loss 计算与梯度更新组成. 嵌入层将使用预训练模型对自然语言进行编码, 获取文本特征, 传入特征融合层; 特征融合层将对获取的文本特征进行进一步处理, 使用双路并行网络学习, 在长路径使用自编码器融合层级特征, 之后使用自注意力模块融合长路径与短路经特征, 随后传入更新层; 而更新层将得到 batch 内所有的文本特征, 并根据 batch 内正负例计算相似度并计算 loss 值, 将正例在投影空间中拉进, 将负例推远, 累积梯度, 更新模型.

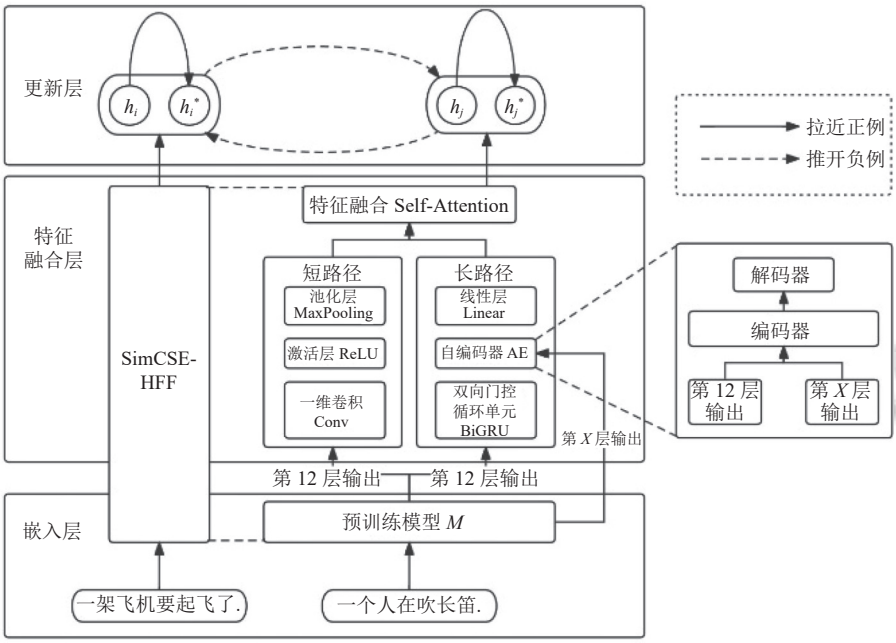


图 2 模型整体架构图

3.2 模型实现流程

3.2.1 嵌入层

嵌入层将使用预训练模型对自然语言进行编码, 获取文本特征, 如图 2 中嵌入层所示.

在具体实现中, 给定 batch 内文本 $S=[s_1,s_2,\cdots,s_i,\cdots,s_n]$, s_i 为 batch 中每一个句子, n 为设定的 batch 句子数量. 输入由 position_embedding, segement_embedding 与 token_embdding 组成. 使用嵌入层 Embedding 将文本集合转换为输入集合.

$$P=\text{Embedding}(S) \tag{1}$$

以 BERT 为例, 模型由 12 层 Transformer 的 encoder 块组成, 进行特征提取. Transformer 块由几个重要的组件组成, 包括多头注意力机制 (multi-head attention)、前馈神经网络 (feed-forward neural network)、残差连接 (residual connection) 和层归一化 (layer normalization). 给定预训练模型, 给定输入 P_i, 在经过本层后得到输出 h_i. 其中 l 代表输出的结果来自于第 l 层. 嵌入层最终输出为 Transformer 块第 l 层的结果.

$$h_i^l=M(P_i) \tag{2}$$

SimCSE 架构将每个语句所对应的 P_i 输入预训练模型 M 两次, 通过 Dropout 获得两个相似向量 h_i 与 h_i^*作为正例对, 而 batch 内其他样例为负例.

3.2.2 特征融合层

特征融合层将对获取的文本特征进行进一步处理, 该层基于双路并行网络并设计长短路径, 短路径使用卷积神经网络学习文本局部特征并进行降维, 长路径使用双向门控循环神经网络学习深度语义信息并利用自编码器融合模型内部其他层特征, 然后使用 Self-Attention 自注意力模块融合双路特征. 具体结构如图 2 特征融合层所示.

在具体实现中, 嵌入层输出 Transformer 块第 l 层和最后一层的结果后, 短路径只使用最后一层结果. 短路径首先使用一维卷积神经网络 $\text{Conv}_{1\text{d}}$, 对输入的文本向量在特征维度方向上进行卷积操作, 压缩特征维度.

$$h_{ia}^{12}=\text{Conv}(h_i^{12}) \tag{3}$$

使用 ReLU 激活函数, 随后进入池化层, 在输入的长度方向上进行最大值池化操作, 得到短路径最后的特征 h_{ia}.

$$h_{ia}=\text{MaxPooling}(\text{ReLU}(h_{ia}^{12})) \tag{4}$$

长路径则会同时使用第 l 层和最后一层的输出结果. 首先, 使用最后一层的特征, 将其输入 BiGRU, 也就是双向 GRU 网路, 进行更进一步的学习与降维. GRU 使用更新门和重置门来控制信息的流动与更新, 从而

更好地捕捉序列信息. 相对于 LSTM, GRU 只包含更新门和重置门两个门控, 具有更简洁的结构. 这导致 GRU 在计算上更高效, 参数较少, 更易于训练和调整, 被广泛用于 NLP 任务中.

$$h_{ib}^{12} = \text{BiGRU}(h_i^{12}) \tag{5}$$

随后, 得到的特征向量进入 AE 自编码器模块, 将第 l 层的特征和最后一层的特征在特征维度方向上拼接作为 AE 模块的输入, 使用编码器 encoder 学习后由解码器 decoder 重建. encoder 与 decoder 均由全连接层组成. 之后再使用全连接层进行降维, 得到长路径最后的特征.

$$\text{AE}(h_{ib}^{12};h_i^l) = \text{decoder}(\text{encoder}(\text{concat}(h_{ib}^{12},h_i^l))) \tag{6}$$

$$h_{ib} = \text{Linear}(\text{AE}(h_{ib}^{12};h_i^l)) \tag{7}$$

最后将长短路径特征进行特征融合. 模块使用 Self-Attention 自注意力模块进行特征融合. Self-Attention 擅长捕捉特征内部特征, 同时可以解决长距离依赖问题. 将两个特征在长度方向上拼接后放入自注意力模块, 最后得到最终的特征h_i.

$$h_i = \text{Self-Attention}(h_{ia};h_{ib}) \tag{8}$$

3.2.3 更新层

更新层将得到 batch 内所有的文本特征, 并根据 batch 内正负例计算相似度并计算 loss 值, 累积梯度, 更新模型.

本文采用余弦相似度来计算两个特征的相似度并计算损失函数.

$$sim(h_i,h_j) = \frac{h_i \cdot h_j}{\|h_i\| \cdot \|h_j\|} \tag{9}$$

采用 InfoNCE 作为模型的损失函数.

$$l = -\log \frac{e^{sim(h_i,h_i^*)/\tau}}{\sum_{j=1}^{N} e^{sim(h_i,h_j)/\tau}} \tag{10}$$

其中, τ 为温度系数, 本文按照 SimCSE 文献[3]设置为 0.55. 分子为正例对相似度, 分母为正例对与负例对的相似度, 最小化该损失函数就是最大化正例对相似度的同时最小化负例对相似度.

4 实验分析

4.1 实验数据

文本匹配可以基于两文本之间语义相似度来判断二者是否匹配, 因此本文选取无监督语义相似度作为本文的实验. 该任务通过模型得到样本句子对向量, 检验样本句子对之间的语义相似性, 衡量本文方法是否优化了模型生成的文本向量质量. 本文实验数据集采用了 STS-B 中文版和 STS-B 英文版. STS-B 是语义文本相似性基准测试, 来自 GLUE (general language understanding evaluation) 基准. STS-B 训练集有 5231 条数据, 验证集有 1458 条数据, 测试集有 1361 条数据.

4.2 实验评估指标

实验使用 *Spearman* 相关性与 *Pearson* 相关性作为实验效果指标. *Spearman* 相关性用来衡量两个数据集的单调关系, 是否有一致趋势; *Pearson* 相关性用来评估相似性或相关性. 指标计算公式如下:

$$Spearman = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)} \tag{11}$$

$$Pearson = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}} \tag{12}$$

$$d_i = R(x_i) - R(y_i) \tag{13}$$

其中, 等级$R(x_i)$为x_i在预测结果集合$X = [x_1, x_2, \cdots, x_i, \cdots, x_n]$里从小到大的排序. d_i为等级差值, 由预测结果集合$X = [x_1, x_2, \cdots, x_i, \cdots, x_n]$中$x_i$的等级$R(x_i)$和验证集 label 集合$Y = [y_1, y_2, \cdots, y_i, \cdots, y_n]$中的$y_i$的等级$R(y_i)$差值绝对值计算而成.

4.3 模型参数优化

无监督语义相似度任务与检索问答任务都使用同一套参数. 实验使用 SimCSE-HFF 方法, 基于 RoBERTa 基座模型, 通过无监督语义相似度任务在中文 STS 数据集上的多次实验, 本文确定了相关参数. 其中影响较大的为 GRU 层数、自编码器参数与额外融合层层数. GRU 层数显著影响模型大小、参数大小, 影响并行网络效果, 而自编码器和额外融合层层数影响层级特征融合的效果.

4.3.1 门控循环神经网络参数设置

本文测试了 GRU 层数对模型效果与模型大小的影响, 实验结果如图 3 所示. 模型大小随着层数增加而变大, 在 2 层上升至 4 层时模型效果有所提升, 但是在继续上升至 6 层时开始下降, 这是因为过于复杂和过于深的网络影响了整体效果. 本文综合考虑模型大小与模型效果后, 最终选择 4 层 GRU 作为最终参数.

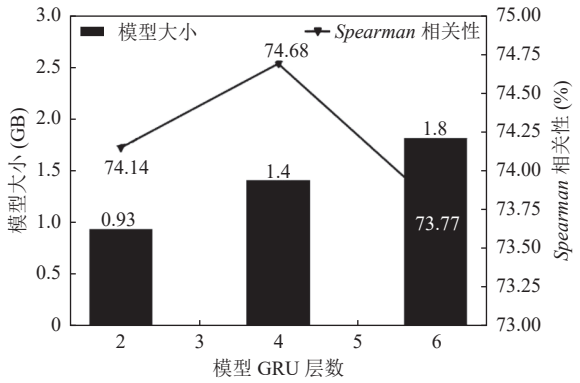


图 3 GRU 层数影响

4.3.2 自编码器维度设置

自编码器内部维度对应隐层单元数量, 不同数量变化对效果有不同影响. 本文对比了 4 组自编码器参数组, 具体参数如表 1 所示, 其中, 数值分别为编码器输入维度、编码器中间层维度、编码器输出维度与解码器输入维度、解码器中间层维度、编码器输出维度, ↓箭头代表与参数组 1 相比维度有所下降. 效果变化图如图 4 所示. 实验控制了输入输出, 测试 AE 在解码器编码器中间参数变化效果. 以参数组 1 为对照组, 编码器隐层单元最大, 解码器隐层单元最大. 参数组 2 解码器隐层单元较低, 效果小幅提升. 参数组 3 编码器隐层单元较低, 效果小幅提升. 而参数组 4 则结合前两者, 效果有明显提升. 对比最后结果发现, 隐层单元的数量较大时, 学习的单元组合多, 携带的冗余信息也多. 而参数组 4 削减了数量, 效果也就相应提升.

表 1 AE 参数组

参数名	具体参数
参数组1	1 792, 768, 128, 512, 768
参数组2	1 792, 768, 128, 256↓, 768
参数组3	1 792, 512↓, 128, 512, 768
参数组4	1 792, 512↓, 128, 256↓, 768

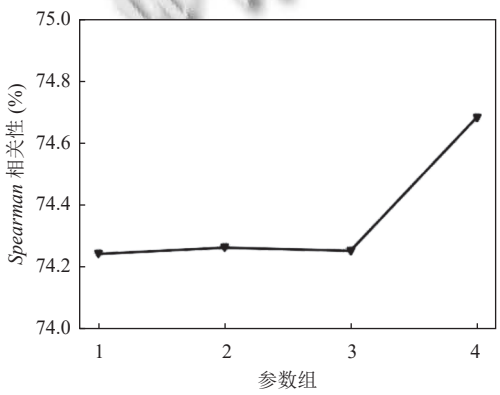


图 4 效果变化图

4.3.3 自编码器特征融合层数设置

实验针对 STS-B 数据集对前 12 层的效果进行了测试, 所有测试结果如图 5 所示. 可以看出, 不同层对模型效果都有不同程度影响, 其中浅层和深层都有不小提升. Jawahar 等[11]的工作也显示深层网络在语义层任务上有更好的效果.

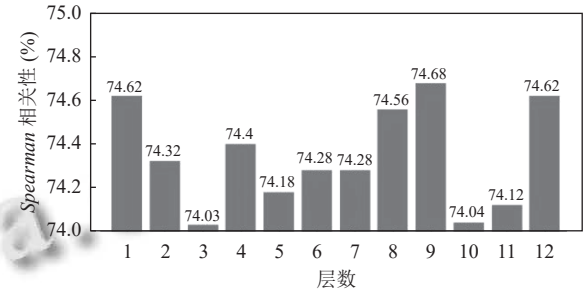


图 5 SimCSE-HFF 前 12 层特征融合效果测试

本文同时进行多层融合实验, 根据以上实验效果从高到低选择了第 9 层、第 1 层、第 8 层, 分别对比了 (1) 9 层、(2) 1+9 层、(3) 1+8+9 层 3 个实验, 实验结果如图 6 所示. 对比结果发现, 在增加融合层数时效果下跌. 本文推测是因为增加层数将显著增加 AE 编码器输入维度, 从而导致 AE 模块效果不佳. 因此本文选择单层融合, 并选择最佳效果第 9 层作为额外特征.

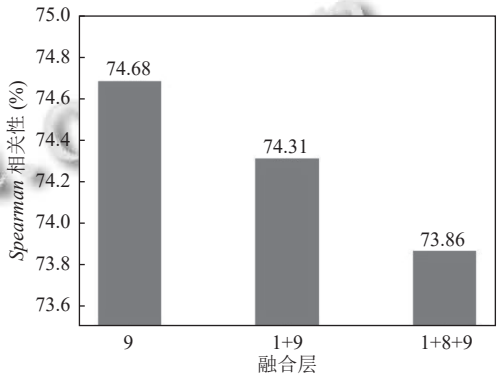


图 6 SimCSE-HFF 多层特征融合效果测试

4.3.4 整体参数设置

本文在优化了以上 3 个参数后, 根据实验条件与论文经验[3]设定了学习率、BatchSize、输入最大长度、CNN 卷积核大小、BiGRU 层数等参数项, 所有实验相关的参数设置如表 2 所示.

4.4 无监督语义相似度实验结果

无监督语义相似度实验使用数据集 STS-B 中文和英文版, 使用余弦相似度计算二者的相似度, 并计算

Spearman 相关性与 *Pearson* 相关性. 针对不同框架, 本文选择了孪生网络 SBERT、对比学习 SimCSE, 与本文提出的 SimCSE-HFF 进行比较. SBERT 为工业界常用框架, SimCSE 为本文改进的源框架, 希望对比这些框架以验证本文方法的有效性. 另外, 实验针对不同预训练模型 BERT 和 RoBERTa 两种模型进行对比, 以及针对中文和英文进行实验, 以验证本文方法的通用性. BERT 是一种基于 Transformer 的预训练语言表示模型, 旨在为自然语言处理任务提供高质量的特征表示. RoBERTa 是 BERT 的一个改进版本, 它通过一系列的优化策略来提高 BERT 的性能.

表 2 参数设置

参数项	数值
学习率	0.000 01
BatchSize	16
输入最大长度	200
CNN卷积核大小	2
BiGRU层数	4
BiGRU输出维度	512
AE参数组	1 792, 512, 128, 256, 768
额外融合层层数	9
温度系数τ	0.55

以 SimCSE-RoBERTa-HFF 方法为例, 语义相似度训练过程 *Spearman* 相关性与 *Pearson* 相关性变化图与模型训练 loss 收敛如图 7 与图 8. 如图 7 所示, 模型收敛迅速, 模型性能在短时间内得到提升, 同时在训练过程中无明显的大幅度波动, 证明了本文方法的稳定性. 相关实验所有结果如表 3 所示.

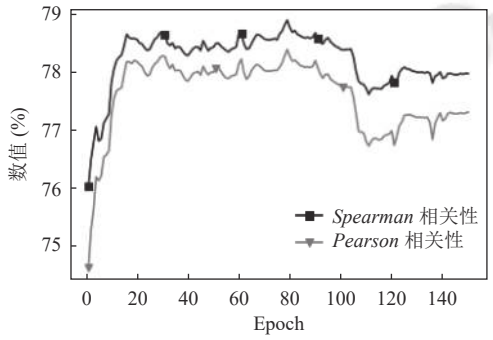


图 7 *Spearman* 相关性与 *Pearson* 相关性变化图

如表 3 所示, SimCSE-BERT-HFF 在中文 *Spearman* 相关性上提升了 0.10 但在 *Pearson* 相关性上低于最优 0.23; 在英文 *Spearman* 相关性、*Pearson* 相关性上提升了 2.65 和 2.18. 而 SimCSE-RoBERTa-HFF 在中文 *Spearman* 相关性、*Pearson* 相关性上提升了 1.00 和 0.96, 在英文 *Spearman* 相关性上提升了 0.36 但在 *Pearson* 相关性上低于最优 0.69.

在针对框架的对比时, 与基础的 SimCSE 方法相比, SBERT 相对较差, 本文提出的 SimCSE-HFF 方法取得了较好的成绩, 只在 *Pearson* 相关性指标下略逊于中文 SimCSE-BERT 和英文 SimCSE-RoBERTa.

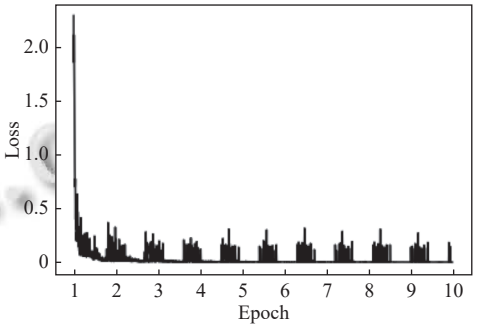


图 8 Loss 变化图

表 3 语义相似度实验结果 (%)

方法	STS-B 中文		STS-B 英文	
	Spearman 相关性	*Pearson* 相关性	*Spearman* 相关性	*Pearson* 相关性
SBERT-BERT	69.92	71.62	72.31	72.48
SimCSE-BERT	71.15	**71.82**	71.67	73.09
SimCSE-BERT-HFF	**71.25**	71.59	**74.32**	**75.27**
SBERT-RoBERTa	70.72	72.58	71.99	72.76
SimCSE-RoBERTa	73.68	74.41	78.33	**79.93**
SimCSE-RoBERTa-HFF	**74.68**	**75.37**	**78.69**	79.24

针对预训练模型的对比, RoBERTa 版本 SimCSE-HFF 在中文上取得了最好的成绩, 在英文上 *Spearman* 相关性指标优于 SimCSE 而 *Pearson* 相关性指标率低于 SimCSE. 而 BERT 版本 SimCSE-HFF 在英文上取得了最好的成绩, 在中文上 *Pearson* 相关性略低于 SimCSE. 本文推测预训练模型基础质量对实验依旧有影响, RoBERTa 模型相较于 BERT 模型拥有更优秀的文本表征能力, 不会出现严重的各向异性.

针对中文和英文的对比, SimCSE-HFF 在中文与英文上都获得了提升.

综上所述, SimCSE-HFF 方法在框架、预训练模型和语种 3 个维度上都获得了优秀的结果, 这体现了本文方法的有效性和通用性.

4.5 消融实验

为了验证 SimCSE-HFF 提出的双路并行网络和层级信息特征融合的有效性, 本文采用 4 个实验对比, 基

座模型 RoBERTa, 分别是 (1) SimCSE-RoBERTa-HFF 本文方法. (2) SimCSE-RoBERTa-HFF 去除 AE 模块. (3) SimCSE-RoBERTa-HFF 去除 AE 模块中第 *l* 层特征融合. (4) SimCSE 原框架. 实验 (2) 将验证双路并行网络对特征进行进一步学习的有效性, 而实验 (3) 将证明第 *l* 层特征融合的有效性. 消融实验的实验结果如表 4 所示.

表 4 中文 STS-B 上消融实验结果 (%)

方法	*Spearman* 相关性	*Pearson* 相关性	*Spearman* 相关性优化	*Pearson* 相关性优化
SimCSE-RoBERTa-HFF	**74.68**	**75.37**	↑1.0	↑0.96
–AE模块	74.41	75.17	↑0.73	↑0.76
–第*l*层特征融合	74.14	74.82	↑0.56	↑0.41
SimCSE-RoBERTa原框架	73.68	74.41	—	—

如表 4 所示, 在去除了 AE 模块、去除第 *l* 层特征融合两种情况下, 相较于 SimCSE-RoBERTa 原框架都有不同程度的效果提升.

去除 AE 模块实验中, 模型只保留双路并行网络, 相较于 SimCSE-RoBERTa 原框架依旧拥有 0.97 的提升, 这说明双路并行网络模块学习到了更多的特征.

在去除第 *l* 层特征融合实验中, 效果相较于去除 AE 模块有所下降. 本文推测是因为 AE 自身的特性导致. AE 对于输入进行压缩后再升维, 通常被用于生成. 而在去除第 *l* 层特征融合方法中, 如果去除了第 *l* 层拼接, AE 退化为一个简单的还原过程, 反而拖累了模型, 这也体现了融合层级特征的重要性.

4.6 下游任务实验

为了验证 SimCSE-HFF 在下游任务的能力, 本文进行了检索问答实验, 通过测试正确答案是否在模型根据语义相似度召回的答案集中, 衡量模型得到的文本向量的质量.

如图 9 所示, 模型训练完成后, 在下游任务使用时只使用预训练模型来生成文本向量, 不使用特征融合层与更新层.

实验使用 QualityQA 数据集来测试模型在下游任务中的性能. QualityQA 为本文收集的企业综合质量诊断专家问答数据, 由企业提出自身问题后, 专家进行回答, 总共 281 条数据. 由于部分实验为无监督学习, 且 QualityQA 数据较少, 所以针对 QualityQA 的实验, 采取 STS-B+QA 的方式, 将 QualityQA 中训练集的 query 与 answer 加入 STS-B 的训练集中, 而在测试中使用测试集, 实验将计算 query 与所有 answer 的相似度并进行召回.

将评级指标设置为 Top5 与 Top10, 指标代表前 5 个召回和前 10 个召回中是否有答案. 相关实验结果如表 5 所示.

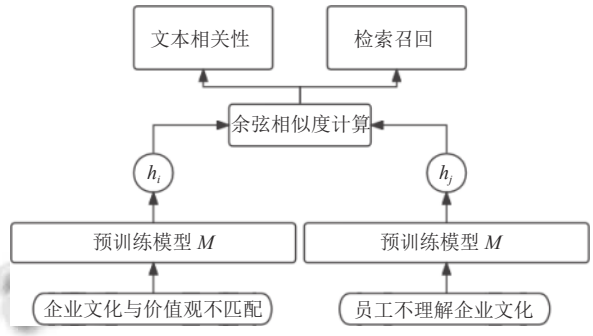


图 9 模型使用图

表 5 QualityQA 数据集上检索问答实验结果 (%)

方法	Top5	Top10
SimCSE-RoBERTa	63.60	75.60
SimCSE-RoBERTa-FF	**65.00**	**75.90**

如表 5 所示, 在 RoBERTa 版本中, SimCSE-RoBERTa-FF 方法将 Top5 与 Top10 提升到了 65.00% 和 75.90%, 分别提升了 1.40% 和 0.30%. SimCSE-RoBERTa-FF 方法更准确地捕捉到了输入句子的语义信息和关键特征. 通过提升文本向量的质量, 模型更好地理解了问题和答案之间的关系, 从而更精确地召回答案.

5 结论与展望

本文针对 SimCSE 框架仅使用[CLS]token 作为文本向量, 对基座模型内层级信息利用不充分的问题, 提出一种融合预训练模型层级特征方法 SimCSE-HFF. SimCSE-HFF 基于双路并行网络, 使用短路径和长路径, 短路径使用卷积神经网络学习文本局部特征并进行降维, 长路径使用双向门控循环神经网络学习深度语义信息, 解决模型对输出特征提取不充分的问题. 同时, 在长路径中利用自编码器 AE 融合模型内部其他层特征, 获取更丰富的特征信息, 解决基座模型内层级信息利用不充分问题. SimCSE-HFF 提高了文本向量生成质量, 更好地理解问题和文本之间的关系, 在 STS-B 的中文与英文数据集上 *Spearman* 和 *Pearson* 相关性指标上优于传统方法, 在不同预训练模型上均得到提升, 同时提高了下游检索任务准确度. 本文在研究中同时提出了一些问题, 例如在引入新结构后, 训练速度的

消耗增大问题, 以及后处理是否会对整体参数分布的一致性与均匀性产生影响. 这些问题也是未来研究中需要更多注意的事项.

参考文献

1 曹帅. 基于深度学习的文本匹配研究综述. 现代计算机, 2021, 27(16): 74–78. [doi: 10.3969/j.issn.1007-1423.2021.16.016]

2 Peters ME, Neumann M, Iyyer M, *et al*. Deep contextualized word representations. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. New Orleans: ACL, 2018. 2227–2237.

3 Devlin J, Chang MW, Lee K, *et al*. BERT: Pre-training of deep bidirectional Transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis: ACL, 2019. 4171–4186.

4 Cai XY, Hang JJ, Bian YC, *et al*. Isotropy in the contextual embedding space: Clusters and manifolds. Proceedings of the 9th International Conference on Learning Representations. ACL, 2021. 1–22.

5 Gao TY, Yao XC, Chen DQ. SimCSE: Simple contrastive learning of sentence embeddings. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Punta Cana: ACL, 2021. 6894–6910.

6 Khattab O, Zaharia M. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. Xi'an: ACM, 2020. 39–48.

7 Humeau S, Shuster K, Lachaux M A, *et al*. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. Proceedings of the 8th International Conference on Learning Representations. Addis Ababa: ICLR, 2020. 1–14.

8 Vaswani A, Shazeer N, Parmar N, *et al*. Attention is all you need. Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach: ACM, 2017. 6000–6010.

9 Lan ZZ, Chen MD, Goodman S, *et al*. ALBERT: A lite BERT for self-supervised learning of language representations. Proceedings of the 8th International Conference on Learning Representations. Addis Ababa: ICLR, 2020. 1–17.

10 Lewis M, Liu YH, Goyal N, *et al*. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Florence: ACL, 2020. 7871–7880.

11 Jawahar G, Sagot B, Seddah D. What does BERT learn about the structure of language? Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence: ACL, 2019. 3651–3657.

12 Chen Q, Zhu XD, Ling ZH, *et al*. Enhanced LSTM for natural language inference. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver: ACL, 2017. 1657–1668.

13 Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation, 1997, 9(8): 1735–1780. [doi: 10.1162/neco.1997.9.8.1735]

14 Pang L, Lan YY, Guo JF, *et al*. Text matching as image recognition. Proceedings of the 30th AAAI Conference on Artificial Intelligence. Phoenix: ACM, 2016. 2793–2799.

15 李广, 刘新, 马中昊, 等. 融合多角度特征的文本匹配模型. 计算机系统应用, 2022, 31(7): 158–164. [doi: 10.15888/j.cnki.csa.008544]

16 后琦, 陈籽健, 刘璐. 结合意图与关键词的句子级交互文本匹配模型. 信息技术与信息化, 2023(10): 24–29. [doi: 10.3969/j.issn.1672-9528.2023.10.005]

17 Huang PS, He XD, Gao JF, *et al*. Learning deep structured semantic models for Web search using clickthrough data. Proceedings of the 22nd ACM International Conference on Information & Knowledge Management. San Francisco: ACM, 2013. 2333–2338.

18 Shen YL, He XD, Gao JF, *et al*. A latent semantic model with convolutional-pooling structure for information retrieval. Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management. Shanghai: ACM, 2014. 101–110.

19 Wan SX, Lan YY, Guo JF, *et al*. A deep architecture for semantic matching with multiple positional sentence representations. Proceedings of the 30th AAAI Conference on Artificial Intelligence. Phoenix: ACM, 2016. 2835–2841.

20 Reimers N, Gurevych I. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hong Kong: ACL, 2019. 3982–3992.

21 Kim T, Yoo KM, Lee SG. Self-guided contrastive learning for BERT sentence representations. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. Bangkok: ACL, 2021. 2528–2540.

(校对责编: 孙君艳)