计算机科学
COMPUTER SCIENCE

基于改进自注意力机制和表示学习的分层文档分类方法

廖兴滨, 钱杨舸, 王乾垒, 秦小林

引用本文

廖兴滨, 钱杨舸, 王乾垒, 秦小林. 基于改进自注意力机制和表示学习的分层文档分类方法[J]. 计算机科学, 2024, 51(2): 238-244.

LIAO Xingbin, QIAN Yangge, WANG Qianlei, QIN Xiaolin. Hierarchical Document Classification Method Based on Improved Self-attention Mechanism and Representation Learning [J]. Computer Science, 2024, 51(2): 238-244.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.221100266

基于改进自注意力机制和表示学习的分层文档分类方法

廖兴滨　钱杨舸　王乾垒　秦小林
中国科学院成都计算机应用研究所自动推理实验室　成都 610213
中国科学院大学计算机科学与技术学院　北京 100080
(liaoxingbin20@mails.ucas.ac.cn)

摘　要　文档分类的一项基本工作是研究如何高效地表示输入特征,句子和文档向量表示也可以辅助自然语言处理的下游任务,如文本情感分析和数据泄露预防等。特征表示也逐渐成为文档分类问题的性能瓶颈和模型可解释性的关键之一。针对现有分层模型面临的大量重复计算以及可解释性缺乏的问题,提出了一种分层文档分类模型,并研究了句子和文档表示方法对文档分类问题的性能影响。所提模型集成了使用改进自注意力机制融合输入特征向量的句子编码器和文档编码器,形成了一个层次结构,以实现对文档级数据的分层处理,在简化计算的同时增强了模型的可解释性。与仅使用预训练语言模型的特殊标记向量作为句子表示的模型相比,所提模型在5个公开文档分类数据集上实现了平均4%的性能提升,比使用词向量矩阵的注意力输出均值的模型提高了2%。

关键词:句子表示;文档表示;注意力机制;文档分类;模型可解释性
中图分类号　TP183

Hierarchical Document Classification Method Based on Improved Self-attention Mechanism and Representation Learning

LIAO Xingbin,QIAN Yangge,WANG Qianlei and QIN Xiaolin
Laboratory for Automated Reasoning and Programming,Chengdu Institute of Computer Applications,CAS,Chengdu 610213,China
School of Computer Science and Technology,University of Chinese Academy of Sciences,Beijing 100080,China

Abstract　An essential task of document classification is to study how to effectively represent input features,and sentence and document vector representations can assist in downstream tasks in natural language processing,such as text sentiment analysis and data leakage prevention. Feature representation is also increasingly becoming one of the keys to performance bottlenecks and interpretability of document classification problems. A hierarchical document classification model is proposed to address the problems of extensive repetitive computation and lack of interpretability faced by existing hierarchical models,and the performance effects of sentence and document representations on the document classification problem are investigated. The proposed model integrates a sentence encoder and a document encoder that fuses input feature vectors using an improved self-attention mechanism,forming a hierarchy to enable hierarchical processing of document-level data,simplifying the computation while enhancing the interpretability of the model. Compared with the model that only uses the special token vector of pre-trained models as sentence representation,the proposed model can achieve an average of 4% performance improvements on five public document classification datasets,and an average of about 2% higher than the model that uses mean attention outputs of word vector matrix.

Keywords　Sentence representation,Document representation,Attention mechanism,Document classification,Model interpretability

1　引言

文档分类模型旨在提取整个上下文的特征,并根据这些特征对文档进行分类。句子表示(Sentence Representation),又称句子嵌入(Sentence Embedding),是一种把句子映射成高维空间中的向量表示的编码技术,即使用一个高维向量表示整个句子,从而帮助提升神经网络处理文本数据的性能。文档表示(Document Representation)则是以文档为单位进行编码,使用高维向量表示文档。文档分类任务则是对文档的表示向量进行分类,得到其类别的概率分布。

到稿日期:2022-11-30　返修日期:2023-04-06
基金项目:四川省科技计划(2019ZDZX0006,2020YFQ0056);中科院STS计划区域重点A类(KFJ-STS-QYZD-2021-21-001)
This work was supported by the Sichuan Science and Technology Program(2019ZDZX0006,2020YFQ0056) and Science and Technology Service Network Initiative(KFJ-STS-QYZD-2021-21-001).
通信作者:秦小林(qinxl2001@126.com)

自然语言处理(Natural Language Processing，NLP)领域的研究人员对文本的编码技术进行了大量研究。词嵌入方法是文本数据处理的基础，也是所有下游任务首要考虑的问题，因为计算机程序无法理解并处理文本数据。随着对文本处理的规模的增大，一些下游任务逐渐需要更高层次的嵌入方式，如文档分类、篇章分析等，句子嵌入和文档嵌入也得到了学术界大量的研究。许多句子表示方法在下游任务和相应的数据集上都表现出了非常好的性能，并取得了巨大的成功，如Doc2Vec[1]和Sent2Vec[2]。

BERT[3]等一系列预训练语言模型(Pre-trained Language Models，PLMs)的提出为句子表示提供了灵感，一些模型开始借助大量有标签或者无标签的数据训练出基于相似度度量的句子表示模型和文档表示模型，如Sentence-BERT[4]和MultilingualUSE[5]。

考虑到训练超大模型的训练成本高、周期长，以及大量BERT系列PLMs的便捷、公开获取等因素，利用预训练语言模型的词向量和特征融合方法来间接地生成句向量是一个较好的选择，常见的做法包括直接使用PLMs的[CLS]位对应的向量表示，如DocBERT[6]，或者使用平均词向量作为句子表示，同时结合更多文本相关的信息，如文献[7]。

很多特殊领域(如数据泄露预防、可解释性情感计算等)的研究者希望模型能够在识别准确率较高的情况下给出所做决策的依据，而很多模型往往只关注模型的准确率指标，难以就当前结果给出合理解释。在自注意力机制[8]的基础上，本文提出了改进的自注意力机制以及基于改进自注意力机制的句子和文档表示方法，借助下游任务训练出句子编码器和文档编码器，可以用于文档分类等下游任务。本文的主要贡献如下：

1)提出了一种分层注意力模型(Hierarchical Attention-enhanced Model，HAM)，用于解决文档分类问题，同时关注句子表示和文档表示。所提模型在模型结构的设计和注意力权重的可视化方面都具有很高的可解释性。

2)受计算资源限制，所提模型仅使用改进的自注意力和Bi-LSTM[9]来构建低成本的分类模型。改进后的自注意力机制可以简化两次矩阵乘法运算。

3)使用Bi-LSTM隐式提取时序相关的文本特征，用于修正编码器生成的向量。

2 相关研究

早在2016年，为解决TextCNN[10]丢失文本结构信息问题而提出了HAN[11]模型。该模型采用分层结构和注意力机制，Yang等认为借助注意力机制可以发现句子中不同词的重要程度，因此句子向量可以直接来源于词向量的加权平均，并且可以使用句向量的加权平均来表示文档向量。

HAN模型在长文本分类任务上表现非常好，层次结构的模型非常契合文档的组织结构，使其具有很好的可解释性。在文献[12]中，Ad-Hoc可解释性建模(Interpretable Modeling)观点认为，在可解释性和模型表达能力之间存在权衡，因此有可能找到一个性能强大且可解释的模型。HAN模型就是一个有力的例子，它还证明了注意力机制在NLP中的有效性，并为后续的一系列工作提供了灵感。

随后，在文献[7]中，使用BERT得到的词向量的平均向量和文档的BOW[13]向量进行拼接，作为文档向量进行分类，由于增加了文档的BOW信息，该模型比仅使用平均向量的表示方法性能更优。

由于BERT等预训练语言模型对输入长度的限制(BERT支持的最大输入长度为512)，在对长文档的处理上，PLMs的性能受到了较大限制，因为简单地截断文档容易丢失文本信息，对下游任务不利。文献[14]提出了RoBERT和ToBERT模型，采用切片的方式，将文档切分成片段，分多次送入BERT模型得到词向量，并在BERT后面接入一个循环神经网络(Recurrent Neural Network，RNN)[15]或者Transformer[16]，待最后一个切片也经过RNN或者Transformer后，得到文档表示。

BERT-flow[17]和WhiteningBERT[18]则是对BERT模型输出的词向量做进一步的加工，BERT-flow将BERT模型最后几层的隐状态进行平均以得到句子表示，并将句子表示空间映射到一个标准高斯潜在空间。由于标准高斯分布是各向同性的和凸的，因此语义分布更为光滑。而WhiteningBERT则认为白化(Whitening)操作可有效改善文本的向量表示，通过在BERT的输出层增加一个白化层，就可以显著改善下游任务的性能。此类方法是对预训练模型的词向量进行优化，并启发了对PLMs的词向量进行特征聚合，以得到更高级别的句子表示和文档表示方法。

除HAN使用分层模型进行文档分类外，这种结构也被很多文档分类模型采用。Choi等[19]提出了一个分层模型，分层对句子和文档进行嵌入，他们使用了专用的句子编码器，并使用门控循环单元[20](Gated Recurrent Neural Network，GRU)和注意力机制来聚合句子向量得到文档表示，并将其用于情感分类任务。

CAHAN[21]模型和MFSA-BiLSTM[22]模型分别将分层结构应用于文档分类和情感分类问题，前者使用注意力机制提取深度文档特征，取得了比HAN模型更高的文档分类准确率；后者则使用了多个通道的特征，并使用注意力机制重点加强关键情感信息，以实现更高精度的情感分类性能。

此外，在HAN、文献[19]中的模型、CAHAN和MFSA-BiLSTM等模型中，注意力机制往往是和LSTM配合使用的，这种方式意味着大量重复的计算。为了避免这一劣势，需要将LSTM和注意力机制剥离开来，并且使用新的计算注意力权重向量的方法，因为注意力机制的计算依赖外部信息。自注意力机制则减少了对外部信息的依赖，并且可以并行计算，对自注意力机制的简单改造即可得到注意力权重向量，本文对此展开了研究。

3 模型框架

本文提出的模型由4部分组成，即动态词嵌入模块、句子编码器、文档编码器和分类器，因此模型可分成动态词嵌入层、句子编码层、文档编码层和分类层。该模型的结构如图1所示。

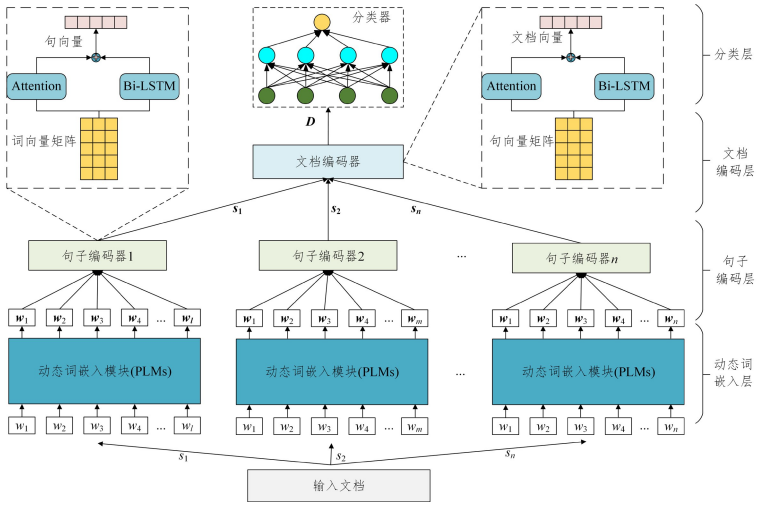

图 1 HAM 模型架构

Fig. 1 Model structure of HAM

分层模型的一个好处是，模型的分层结构对应于文档的组织结构，文档的组织结构是连词成句、构句成篇的，因此按照分层的方式来对词向量和句子向量进行融合是非常合理的。由图 1 可以发现，HAM 模型在动态词嵌入层共享同一个 RoBERTa[23]模块，并且在句子编码层共享同一个句子编码器，也就是说，把文档以句子为单位进行拆分后，将多个句子依次，送入 HAM 模型的 RoBERTa 模块产生词向量，并依次送入句子编码器以产生句向量，随后将所有的句向量拼接成句向量矩阵送入文档编码器得到文档向量，最终由分类层对提取到的文档向量进行分类。

分层文档分类算法的伪代码如算法 1 所示。

算法 1 分层文档分类算法

输入：文档数据训练集 D

输出：分类模型 HAM，注意力权重

1. for $D_i \in D, i=(1,2,\cdots,M)$ do ＃一轮训练
2. 将文档 D_i 以句子为单位进行划分得到句子列表 $S=[s_1, s_2, \cdots, s_n]$；
3. for $s_i \in S, i=(1,2,\cdots,n)$ do
4. 输入到 RoBERTa 模型中，得到词向量矩阵 $\hat{\boldsymbol{s}}_i$，$S=S \backslash s_i$；
5. 句子 s_i 对应的词向量矩阵表示为 $\mathbf{w}=\lambda \hat{\mathbf{s}}_i^{12}+(1-\lambda)\hat{\mathbf{s}}_i^{1}$，$\lambda=0.8$；
6. 将 $\mathbf{w}$ 输入到句子编码器，得到句向量 $\mathbf{s}_i$，句向量矩阵 $\mathbf{S}=\mathbf{S} \cup \mathbf{s}_i$；
7. endfor ＃得到句向量矩阵
8. 将 $\mathbf{S}$ 输入到文档编码器，得到文档向量 $\mathbf{D}_{\text{embed}}$；
9. 将文档向量输入到分类器中，得到分类概率分布并计算分类损失，进行误差反向传导，更新模型参数；
10. endfor ＃一轮训练结束，共训练 20 轮
11. 返回文档分类模型 HAM 和句子编码层，以及文档编码层的注意力权重。

3.1 动态词嵌入模块

如何对文本数据进行编码，以便计算机程序进行处理和计算，是一项必不可少的基础工作，长期以来都是自然语言处理领域的研究热点。词嵌入方法经历了从静态表示到动态表示的演变。BERT 等一系列预训练语言模型是最先进的动态词嵌入模型的代表，可以很好地融合上下文信息，进而动态地对单词进行编码。

通过探测任务[24](Probing Tasks)发现，BERT 的底层捕获了短语级别的信息，底层编码表层特征，中间层编码句法特征，高层编码语义特征。并且 BERT 的表示模仿了经典的树状结构来捕获语言信息。其他工作探索了对下游任务使用不同层的向量表示所带来的性能差异，文献[14]将 BERT 这类模型的第 1 层和第 12 层的隐向量相加并平均作为句子向量。

此外，如果更多地关注某一层捕获的特征，就必须尽量让这一层具有更高的权重。令 $\lambda \in (0,1)$ 表示对第 12 层的权重参数，则词向量矩阵可以表示为：

$$\boldsymbol{w}=\lambda \hat{\boldsymbol{s}}^{12}+(1-\lambda)\hat{\boldsymbol{s}}^{1} \tag{1}$$

其中，$\hat{\boldsymbol{s}}^{12}$ 和 $\hat{\boldsymbol{s}}^{1}$ 分别代表 PLMs 的第 12 层和第 1 层隐藏向量。

3.2 改进自注意力机制

自注意力机制通过由输入特征向量组成并映射得到的查询矩阵 $\boldsymbol{Q}$、键矩阵 $\boldsymbol{K}$ 和值矩阵 $\boldsymbol{V}$ 来进行计算，得到注意力输出：

$$Attention(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=\text{Softmax}\left(\frac{Score(\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}})}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{2}$$

假设一个句子有 d_w 个词，每个词向量的维度为 d_{embed}，其注意力分布矩阵可以计算为：

$$\boldsymbol{M}_{\text{attn}}=\text{Softmax}\left(\frac{Score(\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}})}{\sqrt{d_k}}\right) \tag{3}$$

用 $\alpha_{i,j}=score(\boldsymbol{Q}_i,\boldsymbol{K}_j)$ 表示查询向量 $\boldsymbol{Q}_i$ 和键向量 $\boldsymbol{K}_j$ 的注意力程度，则可将 $\boldsymbol{M}_{\text{attn}}$ 展开为：

$$\boldsymbol{M}_{\text{attn}}=\begin{bmatrix} \frac{\exp(\alpha_{11})}{\sum_{i=1}^{n}\exp(\alpha_{1_i})} & \frac{\exp(\alpha_{12})}{\sum_{i=1}^{n}\exp(\alpha_{1_i})} & \cdots & \frac{\exp(\alpha_{1n})}{\sum_{i=1}^{n}\exp(\alpha_{1_i})} \\ \frac{\exp(\alpha_{21})}{\sum_{i=1}^{n}\exp(\alpha_{2_i})} & \frac{\exp(\alpha_{22})}{\sum_{i=1}^{n}\exp(\alpha_{2_i})} & \cdots & \frac{\exp(\alpha_{2n})}{\sum_{i=1}^{n}\exp(\alpha_{2_i})} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\exp(\alpha_{m1})}{\sum_{i=1}^{n}\exp(\alpha_{m_i})} & \frac{\exp(\alpha_{m2})}{\sum_{i=1}^{n}\exp(\alpha_{m_i})} & \cdots & \frac{\exp(\alpha_{mn})}{\sum_{i=1}^{n}\exp(\alpha_{m_i})} \end{bmatrix} \tag{4}$$

此处的 softmax 操作是按行归一化的。通过对注意力分布矩阵进行简单的处理即可以得到注意力权重向量：第 j 列的平均值表示所有查询向量 $\boldsymbol{Q}_i\ \forall i\in(1,2,\cdots,d_w)$ 对整个序列中第 j 个键向量对应的值向量的平均注意力权重。

因此，将输入序列中第 j 个元素 $\boldsymbol{I}_j$ 的注意力权重表示为：

$$\alpha_j=\frac{1}{d_w}\sum_{i=1}^{d_w}\boldsymbol{M}_{\text{attni}}^{j} \tag{5}$$

对注意力分布矩阵按列取平均可以得到整个序列的注意力权重向量，使用这个权重向量对原始输入序列的特征向量进行加权平均，得到整个序列的向量表示。通过这种方式得到的向量表示有很强的可解释性，从而使得 HAM 模型从设计的角度来看有很强的可解释性：HAM 模型除了使用分层方式来进行句向量和文档向量的聚合外，在句子编码层和文档编码层都利用序列中元素的重要性程度进行了向量表示的提取和优化，得到的文档向量更优。

此外，相比经典的自注意力机制，HAM 模型没有将注意力得分矩阵与值向量矩阵 $\boldsymbol{V}$ 相乘，而是直接使用注意力权重对输入特征矩阵 $\boldsymbol{I}$ 进行加权平均，节省了一次矩阵乘法运算。同时也无须将 $\boldsymbol{I}$ 经过映射得到 $\boldsymbol{V}$ 矩阵，又节省了一次矩阵乘法运算。

3.3 句子编码器

句子编码器使用句子中的词向量来生成句子向量。首先，HAM 模型使用预训练语言模型 RoBERTa 来获取词向量，接着使用改进的自注意力模块获取句子向量。同时，使用 Bi-LSTM 网络从词向量矩阵中提取时序相关的特征，然后将这两个特征相加融合。句子编码器的结构如图 2 所示。

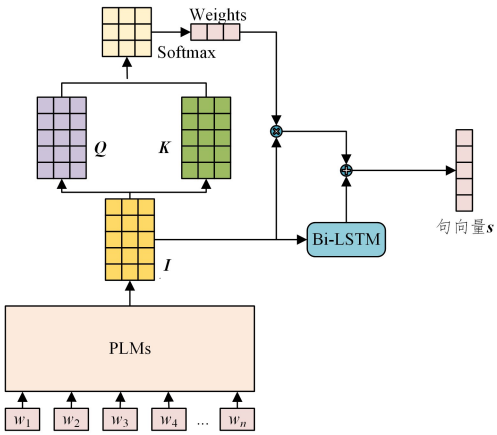

图 2 句子编码器

Fig. 2 Sentence encoder

句子向量可以通过句子中所有词向量的加权平均得到：

$$S_{\text{embed}}=\sum_{i=1}^{d_w}\alpha_i\,\boldsymbol{I}_i \tag{6}$$

其中，$\boldsymbol{I}_i$ 表示输入词向量矩阵中的第 i 个词向量，将式(6)展开后得到式(7)：

$$\boldsymbol{S}_{\text{embed}}=\sum_{i=1}^{d_w}\left(\frac{1}{d_w}\sum_{j=1}^{d_w}M_{\text{senti}}^{j}\right)\boldsymbol{I}_i \tag{7}$$

如果对自注意力机制计算的结果进行平均得到句子表示，则可以表示为：

$$\boldsymbol{S}_{\text{embed}}=\frac{1}{d_w}\boldsymbol{M}_{\text{attn}}^{\text{sent}}\boldsymbol{V} \tag{8}$$

这两种计算句子呈现的方式非常不同。将式(8)作为其中一个基线的句子编码器，并将式(8)作为所提模型的句子向量编码器。一方面，获取单词重要性权重的方式是针对自注意力分布矩阵进行改进的，使得编码器更具有可解释性。另一方面，在训练时将输入矩阵 $\boldsymbol{I}$ 对应的矩阵 $\boldsymbol{V}$ 从计算图中剔除，可以避免矩阵 $\boldsymbol{V}$ 的映射操作。

虽然利用改进的注意力机制可以得到句子中不同词的重要性权重，但是仅仅使用这种加权平均的方法忽略了不同词在序列中的位置关系。当使用单层自注意力机制时，这种位置关系难以用位置编码来表示。于是，使用一个 Bi-LSTM 网络来提取输入词向量矩阵 $\boldsymbol{I}$ 的位置相关的特征，并利用这个特征来修正句子向量：

$$\boldsymbol{S}_{\text{embed}}=\boldsymbol{S}_{\text{embed}}+\boldsymbol{h}_n \tag{9}$$

其中，$\boldsymbol{h}_n$ 表示 Bi-LSTM 网络的最后一个隐状态向量。

3.4 文档编码器

文档编码器使用句子嵌入来生成文档向量。文档编码器中使用的改进的注意力与句子编码器中的注意力相同。在文档编码器中，将注意力权重表示为 $\boldsymbol{\beta}=[\beta_1,\beta_2,\cdots,\beta_n]^{\mathrm{T}}$，其中 β_j 的计算方式参考式(5)。将文档的句向量矩阵表示为 $\boldsymbol{s}=[s_1,s_2,\cdots,s_n]^{\mathrm{T}}$：

$$\boldsymbol{D}_{\text{embed}}=\sum_{i=1}^{d_w}\beta_i\boldsymbol{s}_i \tag{10}$$

本文还使用 Bi-LSTM 网络提取句子编码器生成的句子向量矩阵 $\boldsymbol{S}_{\text{embed}}$ 的位置特征，并利用这个特征来修正文档向量：

$$\boldsymbol{D}_{\text{embed}}=\boldsymbol{D}_{\text{embed}}+\overset{\wedge}{\boldsymbol{h}}_n \tag{11}$$

其中，$\overset{\wedge}{\boldsymbol{h}}_n$ 表示 Bi-LSTM 网络的最后一个隐藏状态向量。

3.5 分类器

在几乎所有的深度学习模型中，全连接网络在整个模型中起到“分类器”的作用。在自然语言处理任务中，上层嵌入层将文本信息转化为词向量，然后通过特征提取模块或表示模块将原始数据映射到隐层特征空间。全连接层融合学习到分布式特征表示并将其映射到样本标签空间。

所提模型使用一个简单的三层全连接网络作为分类器，将文档编码器编码后的向量送入一个维度为 768 的分类器，输出维度为文档数据集的类别数，并使用 softmax 层计算分类的概率分布：

$$p(y=j\mid D)=\frac{\exp(p(D_j))}{\sum_{i=0}^{c}\exp(p(D_i))} \tag{12}$$

其中，$\boldsymbol{D}\in\mathbb{R}^c$ 是文档编码器经过分类层的输出，$p(D_j)\in[0,1]$，c 是分类类别的数量。对于多分类任务，使用交叉熵损失函数：

$$L(x,y)=\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C}\log\frac{\exp(x_{n,c})}{\sum_{i=1}^{C}\exp(x_{n,i})}y_{n,c} \tag{13}$$

所提模型使用的是 PyTorch[25] 提供的交叉熵损失函数，因此式(13)中的损失函数与其一致。

4 实验分析

本章进行综合实验来评估 HAM 在多个文档分类数据集上的性能。

4.1 数据集

本文在 AG_NEWS, DBPedia, IMDB, YelpReviewFull 和 YelpReviewPolarity 这 5 个数据集上测试了 HAM 模型的性能，并与基线模型的性能进行了对比。

在实验开始前，对这些数据集进行了一些统计工作，如表 1 所列。

表 1 5 个数据集的统计信息

Table 1 Statistical information of five datasets

数据集	数据集划分		类别数	句长		平均句子数
	Train	Test		小于 4	大于等于 4	
AG_NEWS	120000	7600	4	118428	1572	1.32
DBPedia	560000	70000	14	464485	95515	2.39
IMDB	25000	25000	2	1929	23071	10.80
YelpReviewFull	650000	50000	5	127952	522048	8.50
YelpReviewPolarity	560000	38000	2	110516	49484	8.49

4.2 基线模型

为了证明所提模型能够在略增加复杂度的情况下提升分类性能，与以下 5 个基线进行了实验比较。

1)CLS：BERT 系列的 PLMs 模型中特殊[CLS]标记的最终隐层向量通常用作句子的表示。本文使用[CLS]向量和文档编码器来组成分类模型，并将这个基线模型命名为 CLS。CLS 模型与 HAM 仅使用的句子编码器不同，因此可以验证 HAM 模型中句子编码器的有效性。

2)mean-embed：一种常用的句子表示方法是对 PLMs 输出的向量进行平均。与使用 CLS 向量一样，这种方法简单有效，几乎没有成本，将该基线模型称为 mean-embed。

3)sentBert：Sentence-bert 是一种专门训练的句子嵌入预训练模型，使用它的输出作为句子向量，并将它与文档编码器结合起来组成基线模型，称为 sentBert。为了确保句子表示方法不受预训练模型的影响，使用与 HAM 中相同的 RoBERTa-base 作为词嵌入层。

4)mean-attn：对预训练语言模型的输出使用经典自注意力机制计算后，输出一个形为 $N\times N$ 的矩阵，其中 N 为输入序列的长度。为了得到句子表示，需要对矩阵进行平均。本文也将其作为句子编码器，与文档编码器结合形成基线模型，称为 mean-attn。

5)HAN：本文也使用 HAN 作为基线。虽然 HAN 是 2016 年提出的，但依旧具有很强的竞争力。使用 GLOVE[26] 作为 HAN 模型的词嵌入层，预训练权重文件为 GLOVE.6B.50d。

4.3 模型配置和训练

所提模型使用 NLTK[27] 提供的分句器(Sentence Tokenizor)将文档切分成多个句子。为了增加模型的泛化能力，实验中没有清洗原始训练数据，简化了模型在实际应用中的使用。

所提模型的句子编码器和文档编码器都使用 Bi-LSTM 网络和改进自注意力机制。将 dropout[28] 率设置为 0.3；分类器使用 3 层全连接层网络，并在每一层使用 Layernorm[29]；将 dropout 率设置为 0.2。通过实验发现，较大的初始学习率有助于加快模型的收敛速度。在模型训练开始时将学习率设置为 0.5，然后在训练过程中每个 epoch 以 0.85 的衰减率降低学习率。使用监督学习来训练所提模型，并使用随机梯度下降[30](Stochastic Gradient Descent，SGD)来优化交叉熵损失。

由于硬件计算资源不足，每个模型在每个数据集上只训练 20 个 epoch。在训练时，冻结了 RoBRETa 的参数，只训练 HAM 模型网络其余部分的参数，未对 PLMs 进行微调。

4.4 实验分析

为了验证本文提出的层次化句子编码和文档编码方法的有效性，在 5 个数据集上进行了训练，并在测试集上进行了性能评估。不同模型在不同数据集上的分类准确率是主要关注的指标。实验结果如表 2 所列，表中数据为 5 次实验数据中记录的最高分类准确率。

表 2 在 5 个数据集上的评估结果(分类准确率)

Table 2 Evaluation results on 5 datasets(accuracy)

(%)

Methods	AG_NEWS	DBPedia	IMDB	YelpReview Full	YelpReview Polarity
CLS	90.23	97.93	90.09	60.54	93.84
mean-embed	91.16	98.32	90.63	61.88	94.67
sentBert	91.79	98.67	83.01	56.74	91.07
mean-attn	92.67	99.06	89.13	65.49	96.72
HAN	91.45	98.64	88.94	60.68	93.76
HAM(Ours)	**94.22**	**99.52**	**94.40**	**67.31**	**97.25**

前 4 个基线模型使用与 HAM 相同的文档编码器，而句子编码器不同。实验结果表明，HAM 在分类准确率上普遍优于基线模型，并且比 CLS 和 mean-embed 平均高出 4.014% 和 3.96%。HAM 模型也比 mean-attn 模型表现更好，分类准确率平均提高了 1.93%。

与 SentenceBERT 相比，HAM 模型在 YelpReviewFull 数据集上的性能有近 7%的提升。尽管 sentBert 的文档编码器使用了改进的自注意力机制，但 SentenceBERT 在长文档训练集上的表现仍然不佳。这也说明了句子表示对分类模型的重要性，因为它可以直接影响文档表示的质量，进而影响分类准确性。

HAN 模型是一个开创性的工作，但与 HAM 模型相比，由于使用了 RoBERTa 词嵌入层和改进的自注意力机制构建的编码器，所提模型的分类准确率比 HAN 高 3.85%。

在所有模型使用相同的文档编码器的前提下，通过比较不同句子编码器引起的性能差异，证明了句子编码器的有效性。

对于数据集中文档的平均句子数量大于 8 的 YelpRe-

viewFull，YelpReviewPolarity 和 IMDB，所提模型相比其他几个基线模型有显著优势，这也表明所提模型对较长的文本非常有效。

4.5 消融实验

HAM 模型使用了改进的自注意力机制和 Bi-LSTM 进行特征融合，移除 Bi-LSTM 后，比较 $\text{HAM}_{\text{no\_lstm}}$ 在 5 个数据集上的性能差异。在 YelpReviewFull 数据集上有 1%的性能损失，这也说明加入修正信息有助于提高模型的准确率。实验结果如表 3 所列。

表 3 与未使用 LSTM 的 HAM 的性能对比

Table 3 Performance comparison of HAM without LSTM

(%)

Methods	AG_NEWS	DBPedia	IMDB	YelpReview Full	YelpReview Polarity
$\text{HAM}_{\text{no\_lstm}}$	93.83	99.13	93.79	66.26	96.98
HAM	**94.22**	**99.52**	**94.40**	**67.31**	**97.25**

此外，前 4 个基线模型中使用了与 HAM 模型相同的文档编码器，为了证明所提模型的文档编码器的有效性，用句子向量矩阵的平均替换文档编码器，并观察替换后的模型在 5 个数据集上的性能，如表 4 所列。

表 4 文档编码器的消融实验

Table 4 Ablation experiment of document encoder

(%)

Methods	AG_NEWS	DBPedia	IMDB	YelpReview Full	YelpReview Polarity
CLS_{mean}	89.21	93.03	87.34	52.61	92.17
CLS	**90.23**	**97.93**	**90.09**	**60.54**	**93.84**
$\text{mean-embed}_{\text{mean}}$	90.83	98.00	89.21	59.13	93.93
mean-embed	**91.16**	**98.32**	**90.63**	**61.88**	**94.67**
$\text{sentBert}_{\text{mean}}$	89.99	98.56	82.78	54.62	89.92
sentBert	**91.79**	**98.67**	**83.01**	**56.74**	**91.07**
$\text{mean-attn}_{\text{mean}}$	92.24	98.50	87.96	63.46	96.23
mean-attn	**92.67**	**99.06**	**89.13**	**65.49**	**96.72**

用句子向量矩阵的平均替换基线模型的文档编码器后，所有模型的性能都出现了下降，这表明所提模型的文档编码器有助于提高分类精度。

综上，消融实验证明了使用改进的自注意力机制构建的句子编码器和文档编码器对句子表示和文档表示的有效性，从而提升了模型的分类性能。

4.6 案例分析

对 HAM 模型不同层的注意力权重的可视化表明，所提模型可以发现句子和文档中不同元素的重要性差异，对其进行的可视化分析证明了 HAM 模型具有较强的可解释性。

以 IMDB 数据集中的一个消极情绪类样本为例，句子编码层和文档编码层的注意力权重可视化结果如图 3 所示。图 3 中红色标记的词或者句子有更大的注意力权重，紫色次之，而没有标记颜色的词则表明其注意力权重小于平均注意力权重，对分类结果的影响也很小。比如在第一个句子中，“worst”这个词带有非常强烈的情感色彩，因此这个词的权重在该句子中是最大的。再比如，在第六句中，“just horrible”成为句子中权重最大的短语。此外，所提模型还捕获了不同句子重要性的差异，如 sent0，sent2 和 sent4，这几个句子的注意力权重更大0，对分类结果的影响也更大。

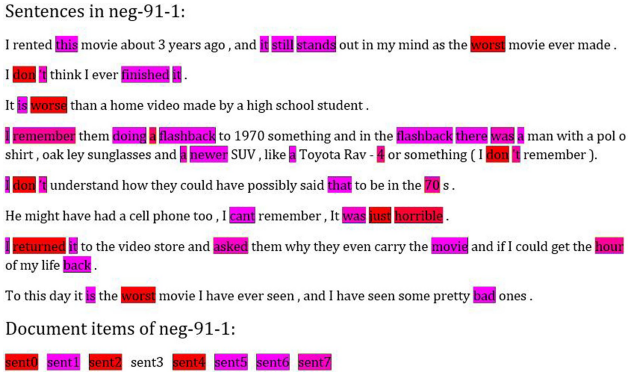

图 3 注意力权重可视化(电子版为彩图)

Fig. 3 Visualization of attention weights

结束语 本文使用改进的注意机制和预训练的语言模型以及 Bi-LSTM 模块构建了分层文档分类模型，研究了模型中有助于提高分类任务性能的具体部分。在 5 个公共数据集上进行了完整的实验，并探索了这种方法与其他基线模型之间的性能差异。实验结果表明，使用改进的注意力机制组成的编码器有助于句子表示和文档表示，有助于下游分类任务的性能提升。所提分类模型具有较高的准确性和良好的可解释性。由于使用 Bi-LSTM 网络对分类准确率的提升并不显著，同时也使模型训练的时间增加了两倍，未来的工作将考虑使用更好的网络结构来提取序列的位置信息，在缩短模型训练时间的基础上，提升了模型的分类性能。

参 考 文 献

[1] LE Q，MIKOLOV T. Distributed representations of sentences and documents[C]// International Conference on Machine Learning. PMLR，2014：1188-1196.

[2] PAGLIARDINI M，GUPTA P，JAGGI M. Unsupervised Learning of Sentence Embeddings Using Compositional n-Gram Features[C]// Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics：Human Language Technologies，Volume 1(Long Papers). 2018：528-540.

[3] DEVLIN J，CHANG M W，LEE K，et al. BERT：Pre-training of Deep Bidirectional Transformers for Language Understanding [C]// Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics：Human Language Technologies，Volume 1(Long and Short Papers). 2019：4171-4186.

[4] REIMERS N，GUREVYCH I. Sentence-BERT：Sentence Embeddings using Siamese BERT-Networks[C]// Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing(EMNLP-IJCNLP). 2019：3982-3992.

[5] YANG Y，CER D，AHMAD A，et al. Multilingual Universal Sentence Encoder for Semantic Retrieval[C]// Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics：System Demonstrations. 2020：87-94.

[6] ADHIKARI A，RAM A，TANG R，et al. Docbert：Bert for docu-

ment classification [EB/OL]. (2019-04-17) [2019-08-22]. https://arxiv.org/abs/1904.08398.
[7] TANAKA H, SHINNOU H, CAO R, et al. Document classification by word embeddings of bert[C]// International Conference of the Pacific Association for Computational Linguistics. Springer, Singapore, 2019: 145-154.
[8] LUONG M T, PHAM H, MANNING C D. Effective Approaches to Attention-based Neural Machine Translation[C]// Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. 2015: 1412-1421.
[9] LI J, XU Y, SHI H. Bidirectional LSTM with hierarchical attention for text classification[C]// 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). IEEE, 2019, 1: 456-459.
[10] CHEN Y. Convolutional neural network for sentence classification[D]. Ontario: University of Waterloo, 2015.
[11] YANG Z, YANG D, DYER C, et al. Hierarchical attention networks for document classification[C]// Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2016: 1480-1489.
[12] FAN F L, XIONG J, LI M, et al. On interpretability of artificial neural networks: A survey[J]. IEEE Transactions on Radiation and Plasma Medical Sciences, 2021, 5(6): 741-760.
[13] HARRIS Z S. Distributional structure[J]. Word, 1954, 10(2/3): 146-162.
[14] PAPPAGARI R, ZELASKO P, VILLALBA J, et al. Hierarchical transformers for long document classification[C]// 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). IEEE, 2019: 838-844.
[15] RUMELHART D E, HINTON G E, WILLIAMS R J. Learning representations by back-propagating errors[J]. Nature, 1986, 323(6088): 533-536.
[16] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]// Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17). Curran Associates Inc., 2017: 6000-6010.
[17] LI B, ZHOU H, HE J, et al. On the Sentence Embeddings from Pre-trained Language Models [C]// Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020: 9119-9130.
[18] HUANG J, TANG D, ZHONG W, et al. WhiteningBERT: An Easy Unsupervised Sentence Embedding Approach[C]// Findings of the Association for Computational Linguistics (EMNLP 2021). 2021: 238-244.
[19] CHOI G, OH S, KIM H. Improving document-level sentiment classification using importance of sentences[J]. Entropy, 2020, 22(12): 1336.
[20] CHO K, VAN M B, GU·LÇEHRE Ç, et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation[C]// Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014: 1724-1734.
[21] LI W J, QI F, YU Z T. Sentiment classification method based on multi-channel features and self-attention[J]. Journal of Software, 2021, 32(9): 2783-2800.
[22] REN J H, LI J, MENG X F. Document classification method based on context awareness and hierarchical attention network [J]. Journal of Frontiers of Computer Science and Technology, 2021, 15(2): 305-314.
[23] LIU Y, OTT M, GOYAL N, et al. Roberta: A robustly optimized bert pretraining approach[EB/OL]. (2019-07-26) [2019-07-26]. https://arxiv.org/ abs/1907.11692.
[24] JAWAHAR G, SAGOT B, SEDDAH D. What Does BERT Learn about the Structure of Language? [C]// Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 3651-3657.
[25] PASZKE A, GROSS S, MASSA F, et al. Pytorch: An imperative style, high-performance deep learning library[J]. Advances in Neural Information Processing Systems, 2019, 32: 8024-8035.
[26] PENNINGTON J, SOCHER R, MANNING C D. Glove: Global vectors for word representation [C]// Proceedings of the 2014 Conference on Empirical Methods in Nnatural Language Processing (EMNLP). 2014: 1532-1543.
[27] BIRD S, KLEIN E, LOPER E. Natural language processing with Python: analyzing text with the natural language toolkit[M]. California: O'Reilly Media, Inc., 2009.
[28] SRIVASTAVA N, HINTON G, KRIZHEVSKY A, et al. Dropout: a simple way to prevent neural networks from overfitting [J]. The Journal of Machine Learning Research, 2014, 15(1): 1929-1958.
[29] BA J L, KIROS J R, HINTON G E. Layer normalization[EB/OL]. (2016-07-21)[2016-07-21]. https://arxiv.org/abs/1607.06450.
[30] ROBBINS H, MONRO S. A stochastic approximation method [J]. The Annals of Mathematical Statistics, 1951, 22(3): 400-407.

LIAO Xingbin, born in 1994, postgraduate, is a member of CCF (No. J9063G). His main research interests include natural language processing and artificial intelligence.

QIN Xiaolin, born in 1980, Ph.D, professor, Ph.D. supervisor, is a senior member of CCF (No. 12344M). His main research interests include artificial intelligence and automated reasoning.

(责任编辑:喻藜)