计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.200800004

基于 Transformer 和多通道卷积神经网络的情感分析研究

霍 帅[1,2] 庞春江[1]
1 华北电力大学(保定) 河北 保定 071003
2 云南电网有限公司电力科学研究院研究生工作站 昆明 650217
(hdhuoshuai@163.com)

摘 要 文本情感分析是自然语言处理的经典领域之一。文中提出了一种基于 transformer 特征抽取器联合多通道卷积神经网络的文本情感分析的模型。该模型使用 transformer 特征提取器在传统 Word2vector,Glove 等方式训练的静态词向量的基础上来进行单词的分层、动态表示,针对特定数据集采用 Fine-Tuning 方式来进行训练有效提升了词向量的表征能力。多通道卷积神经网络考虑了不同大小范围内词序列之间的依赖关系,有效进行特征抽取并达到降维的目的,能够有效捕捉句子的上下文语义信息,使模型捕获更多的语义情感信息,提升文本的语义表达能力,通过 Softmax 激活函数达成情感倾向分类的目标。模型分别在 IMDb 和 SST-2 电影评论数据集上进行实验,测试集上准确率达 90.4%和 90.2%,这明所提模型较传统词嵌入结合 CNN 或 RNN 的模型在分类精确度上有了一定程度的提升。

关键词:情感分类;特征提取器;Transformer;多通道卷积神经网络
中图法分类号 TP391.1

Research on Sentiment Analysis Based on Transformer and Multi-channel Convolutional Neural Network

HUO Shuai[1,2] and PANG Chun-jiang[1]
1 North China Electric Power University(Baoding),Baoding,Hebei 071003,China
2 Graduate Workstation of Yunnan Power Grid Co.,Ltd. Electric Power Research Institute,Kunming 650217,China

Abstract Text sentiment analysis is one of the classic fields of natural language processing. This paper proposes a text sentiment analysis model based on transformer feature extractor combined with multi-channel convolutional neural network. The model uses trsnsformer feature extractor to layer words and dynamically represent them on the basis of static word vectors trained by traditional Word2vector,Glove,etc.,and use Fine-Tuning for specific data sets for training,which effectively improves the representation of word vectors ability. The multi-channel convolutional neural network considers the dependence between word sequences in different size ranges,effectively extracts features and achieves the purpose of dimensionality reduction,can effectively capture the contextual semantic information of sentences,and enable the model to capture more semantic emotional information,improve the semantic expression ability of the text,and achieve the goal of emotional tendency classification through the Softmax activation function. The model is tested on the IMDb and SST-2 movie review datasets,and the accuracy rates on the test set reached 90.4% and 90.2%,indicating that the model proposed in this paper has better classification accuracy than the traditional word embedding combined with CNN or RNN.

Keywords Sentiment classification,Feature extractor,Transformer,Multi-channel convolutional neural network

1 引言

随着互联网时代的到来,人们把自己的感情和看法等通过社交媒体和应用软件等平台发表出来,分析这些信息能够有助于快速掌握评论者的情感倾向,文本情感分析领域得到越来越多的关注。总结目前的国内外相关的研究论文发现文本情感分析的方法大致有 3 类:基于情感词典的方法、基于机器学习的方法和基于深度学习的方法。

在文本情感分析领域,文献[1]认为评论中形容词的极性是判定评论情感极性的主要指标,提出将形容词(如"good""bad"等)作为情感词建立情感词典,再根据词典中情感词的极性来判断评论的情感极性。基于情感词典的方法,需要专家基于语言规则知识构建情感词典,并需要大量的时间和人力成本进行维护更新词典,进而导致领域泛化能力较差。Pang 等[2]首次提出使用标准的机器学习方法解决情感分类问题,该工作对比了不同特征组合与不同机器学习方法在电影评论情感分类问题上的效果。此后,多数机器学习方法的研究工作将重点放在如何设计更多有效的分类特征上,研究者们尝试了不同类特征组合在情感分类上的效果。Saleh 等[3]在 3 个不同数据集上进行了 27 组实验分别测试了不同特征组合方法对情感分类效果的影响。传统机器学习方法把研究重点放在如何设计有效的分类特征上,严重依赖于人工

基金项目:云南科技项目(YNKJXM20180019,YNKJXM20191572)
This work was supported by Yunnan Science and Technology Project(YNKJXM20180019,YNKJXM20191572).
通信作者:庞春江(972158083@qq.com)

设计的特征，易受到人为因素的影响。深度学习取代浅层学习，摆脱了特征工程的束缚，利用语义合成性原理通过不同深度模型将低层词向量合成高层文本情感语义特征向量从而得到文本的高层次情感语义表达，有效提升模型的推广能力。该类方法的效果依赖于数据的质和量、模型的结构、超参的选择以及参数的调节。

Kim 等[4]使用卷积神经网络(Convolutional Neural Network，CNN)对短文本进行建模从而完成了句子级别的文本情感分析任务，使之成为句子级情感分类任务的重要 baseline 之一。文献[5]提出使用长短期记忆网络(Long Short Term Memory，LSTM)来将评论文本建模成词序列以用来解决情感分类的问题。LSTM 与 CNN 提取局部语义特征的特点相比可以捕捉到评论语句中的依赖关系，可以从整体上“理解”评论的情感语义。Zhang 等[6]使用多通道卷积神经网络(Multi-channel Convolutional Neural Network，MCNN)进行局部语义特征信息的提取并结合双向 GRU 模型(Gate Recurrent Unit)来将文本的整体语义信息进行融合从而获得最终的情感倾向。Zhao 等[7]使用预训练字向量初始化 ELMo 的嵌入层之后利用多尺度卷积神经网络进行语义特征提取与融合，从而生成文本句的整体语义表示。

传统情感分析方法普遍侧重于特征工程的特点，导致需要相关从业人员具有较高的领域知识储备。深度学习大大降低了对专业领域知识的需求，近年来飞速发展并实现了一系列的“端到端”模型，其三大特征抽取器 CNN，RNN 以及 Transformer 结构的优缺点各异，其中 CNN 可以捕捉局部特征但远程特征提取能力较弱，RNN 可有效应用于序列模型但存在梯度消失及无法有效捕获长距离依赖的问题，LSTM 和 GRU 虽可通过门控机制较好地处理 RNN 存在的问题却无法并行化。通过预训练语言模型获得的通用语言表征对下游的 NLP 任务有帮助，但也有不足，例如静态词嵌入[8-9](如 Glove，Word2vector)自训练完成后不再变化，当具体应用于下游 NLP 任务时往往与特定上下文文本无关，如果遇到一词多义现象时，效果往往很差。动态词向量如 ELMo[10]其使用 LSTM 抽取文本特征但双向 LSTM 特征拼接方式弱于一体化的双向 transformer[11]。因此本文提出针对特定领域数据集，以静态词嵌入向量为基础使用 transformer 特征抽取器进行语义特征提取，使用 self-attention 解决长距离依赖问题，同时结合 fine-tuning 的方式获取动态词向量表征，并借助多通道卷积神经网络实现特征二次提取及融合的情感分析模型来提升文本语义表达能力。通过在不同数据集上的一系列对比实验验证了本文模型的有效性。

本文主要贡献如下：

使用 transformer 特征抽取器不同层可以捕捉不同语法和语义信息的动态词向量代替传统 Word2vector 或 Glove 等方式训练静态词向量，从而提升了词向量的表证能力，较完整地保存了文本语义信息。

针对特定任务、数据集，引入多通道卷积神经网络对关键信息进行筛选，并通过 fine-tuning 的方式进行模型微调，可以避免繁杂的特征工程从而提升语义的表达能力，使模型更加关注文本中的重要信息，有效提升模型分类准确度，使其有较强的泛化能力。

2 相关工作

深度学习在计算机视觉和自然语言处理领域取得了诸多成果。由于自然语言处理领域文本表示及处理的特殊性，预训练语言模型发展迅速，其应用于自然语言处理领域的思想类似于计算机视觉中的迁移学习。2014 年，Yosinski 等[12]通过实验证了自然图像上训练的深度卷积神经网络每一层特征的普遍性与特异性，较低层神经元提取的特征具有通用性，较高层的特征对特定任务表现出较高相关性，并且通过初始化较低层参数，对目标数据集微调可以增强泛化能力。单词的分布式表示的概念最早由 Hinton 等提出。2013 年，Mikolov 等[8]在 CBOW(continuous bag-of-words)和 Skip-gram 模型的基础上提出了 Word2vec 来获得单词的分布式表示的词向量，同时采用负采样和分层 Softmax 技术进行算法优化，有效减小了算法复杂度，该工作是词嵌入广泛应用于下游 NLP 任务的开端。2014 年，Pennington 等[9]提出的 GloVe 模型通过对单词共现矩阵中的非零元素训练，从而有效地利用全局统计信息并生成有意义的子结构向量空间，也取得了不错的效果。2014 年，Kim 等[4]提出的 TextCNN 模型以 Word2vector 的预训练的词向量作初始化并通过不同大小和数量的卷积核进行不同范围的特征提取与融合，验证了 CNN 应用于文本数据的可行性，也表明了预训练词向量的通用性。文献[13]在 TextCNN 模型的基础提出了改进的 DCNN(Dynamic Convolutional Ceural Network)模型，该模型仍然基于卷积，但采用了宽卷积的思想有效捕获句子边缘信息，对池化层进行了优化，不同于传统地选出最大值，而是动态地选出最大的 k 个值使得提取的文本语义更加完整。文献[14]提出了一种树形的长短期记忆网络模型 Tree-LSTM，与标准 LSTM 隐藏状态由当前输入和前一阶段隐藏状态组成相比，Tree-LSTM 的当前隐藏状态由当前的输入向量和任意个子单元的隐藏状态组成，该模型在语义相关性测试和情感分类中都取得了较好的效果。

2017 年，Vaswani 等[11]提出一种基于 transformer 的 encoder-decoder 架构，编码器和解码器均由 6 个编码 block 组成，其编码器中的 self-attention 结构可在计算当前词的时候同时利用它上下文的词，有效提取词之间长距离依赖关系并且每个 token 表示生成的计算过程都是独立进行的，因此其可并行计算所有 token 特征向量，同时具有抽取长距离依赖关系和并行计算的能力，在多项翻译任务上有效提升了 BLUE 得分。2018 年，Peters 等[10]提出了一种深度语境化词表示法 ELMO，该模型通过给定的任务，使用在大规模数据上预先训练的双向长短期(Bi-LSTM)网络的所有固定隐藏层记忆学习任务特定的加权表示来构建上下文优化的词嵌入，有效解决了多义词问题，在问答、情感分析以及文本蕴含等 6 项 NLP 任务上的效果均有提升。2018 年，Radford 等[15]提出了 GPT(generative pre-training)，该模型在无标注数据上使用 transformer 替代 LSTM(相对而言 LSTM 无法捕捉更长的语义信息)学习语言模型，根据前面的 context 去预测下一个词(只利用了单侧信息)，然后通过有监督的 fine-tune 改进具体的任务，其在文本生成类任务上效果较好。2018 年，Devlin 等[16]提出了 BERT，其通过在预训练过程中调节所有层的上下文来学习深度双向表示。它是一种基于多层双向 transformer 编码在大规模数据上以无监督任务 Masked LM 和 Next Sentence Prediction 为目标预先训练的深层语言表示模型，可以用于微调。通过特定于不同任务的层，它适用于单词和句子级的广泛任务，在阅读理解和文本分类等 11 项

NLP 任务中打破记录。2020 年，谷歌大脑的研究团队[17]通过实验表明，预训练虽可学习通用的表征，但当预训练任务和目标任务存在差异时，预训练可能损害目标任务的准确率，因此针对特定任务和数据集的自监督训练是很有必要的。

综上在短文本分类领域，有效特征（字、词，句级别）表示及充分的语义特征抽取模型（CNN，RNN，Transformer）的提出与改进一直是近年来关注的热点。

3　系统模型

3.1　模型概览 MCNN-Transformer

词嵌入模型（Word2vec 和 Glove）经过预训练过程获得的嵌入向量，可以捕捉单词的语义信息，从而对下游 NLP 任务有较大提升。预训练编码器（ELMO，GPT，BERT）通过一个预训练的编码器根据上下文语境进行动态调整，使得不同层可以捕捉不同的语法和语义信息，从而能够动态生成基于上下文的词向量。因此本文借助其中语义特征提取能力最强的 transformer 特征抽取器来生成动态词向量，针对特定任务和数据集引入多通道卷积神经网络对关键信息进行筛选，并通过 fine-tuning 的方式进行模型微调来提升文本语义表达能力。

本文模型由针对特定领域的动态词向量表征模块和多通道卷积神经网络模型组成，模型整体结构如图 1 所示。

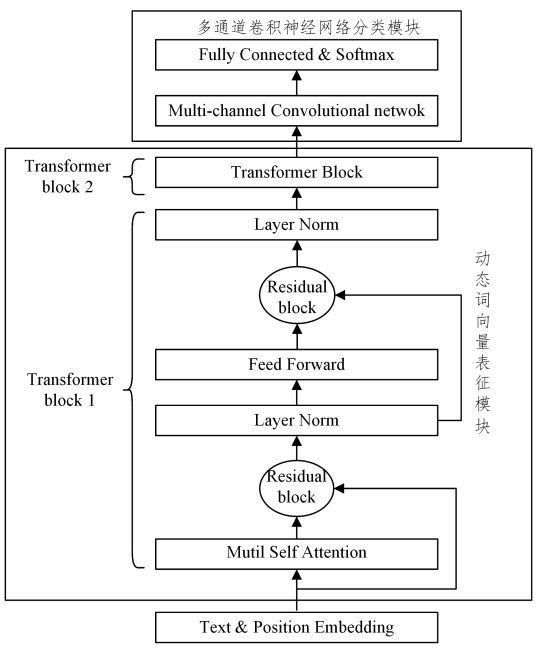


图 1　本文模型 MCNN-Transformer

Fig. 1　Proposed model MCNN-Transformer

模型公式化的表示如下：

$$\boldsymbol{h}_n^0=\boldsymbol{w}_e\boldsymbol{w}_n+\boldsymbol{w}_p \tag{1}$$

$$\boldsymbol{h}_n^l=transformer\_block(\boldsymbol{h}_n^{l-1}) \tag{2}$$

$$y_n^{MCNN\_Transformer}=MCNN(\boldsymbol{w}_0\boldsymbol{h}_n^L+b_0) \tag{3}$$

其中，$\boldsymbol{w}_n$ 代表当前词向量，$\boldsymbol{w}_p$ 代表位置向量，$\boldsymbol{w}_e$ 代表权重向量矩阵，n 表示序列长度，$\boldsymbol{h}_n^0$ 代表输入向量。transformer_block 主要包括自注意力层和前馈神经网络层，$\boldsymbol{h}_n^l$ 代表 transformer_block 的输出，l 代表当前层（$l=1,2$），L 代表模型的总层数。式(3)表示输出层计算过程，以情感二分类任务为例，$y_n^{MCNN\_Transformer}$ 为通过多通道卷积神经网络（Multi-channel Convolutional Neural Network，MCNN）预测属于各个类别的概率的最终输出，$\boldsymbol{w}_0$ 为输出权重矩阵，b_0 为输出层偏差。针对不同数据集使用 fine-tuning 的方式调整参数 w_0，b_0，以及 transformer_block 层参数来最小化误差。我们首先针对特定数据集，使用 transformer 特征提取器进行预训练，获得不同层的动态词向量表示后组合不同层的动态词向量表示进行训练以获得最佳效果。

3.2　动态词向量表征模块

首先两层 *transformer_block* 对词向量（tokenen codeing）与词的位置向量（position encodeing）加权后的文本表征向量进行特征的分层表示，以达到句子特征融合的目的。词向量可以选用在大型语料库中预训练获得的包含更多的先验知识的静态词向量，也可以随机初始化再由当前任务训练生成更好地捕获与当前任务相关联的特征信息的动态向量表征。

3.2.1　位置嵌入和词嵌入

transformer 特征提取器的自注意力模块的结构不能直接获得输入序列的顺序信息，所以需要额外信息来使模型洞悉输入 token 的相对位置（即两个 token 间的相对距离）和绝对位置（当前 token 在整个句子中的位置）信息，即 positional encoding（其中位置向量与 Token Embedding 具有相同维度 d_{model}），将这两个词向量相加后再输入模型。我们采用正弦函数来生成位置嵌入向量：

$$PE(pos,2i+1)=\cos(pos/10\,000^{2i/d_{model}}) \tag{4}$$

其中，pos 代表位置，i 代表维度。

3.2.2　多头注意力机制

点积自注意力（Scaled Dot-Product Attention）模块的计算过程如图 2 所示，式(5)－式(9)是其公式化的表示。

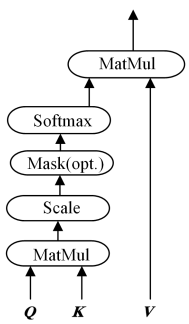


图 2　点积自注意力[11]

Fig. 2　Scaled dot-product attention[11]

针对本文模型，输入为维度 $d_k=36$ 的 query，key 和维度 $d_v=30$ 的 value 向量，其由词向量 a 产生。将 query 分别和每个 key 进行内积运算，并对结果除以 $\sqrt{d_k}$ 缩放后输入 *softmax* 函数，得到权重后乘以 value，获得自注意力层的输出 b。

$$q^i=w^q a^i \tag{5}$$

$$k^i=w^k a^i \tag{6}$$

$$v^i=w^v a^i \tag{7}$$

$$a_{i,j}=q^i\cdot k^j/\sqrt{d_k} \tag{8}$$

$$b^i=\sum_j softmax(a_{i,j})v^j \tag{9}$$

在实际计算中，会将多个 query 打包为矩阵后进行并行计算。key 和 value 也被打包为矩阵 $\boldsymbol{K}$ 和矩阵 $\boldsymbol{V}$，计算过程为式(10)：

$$attention(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{10}$$

Multi-head attention 能够让模型从不同的表征子空间去共同学习不同位置的表达信息。类似于在 CNN 中的多个 filter 学习图片不同的表达信息，如图 3 所示，先将 $\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}$ 经过不同的 h 个线性投影后，再进行 Scaled Dot-Product Attention 的计算，可以学习到不同的语义信息。

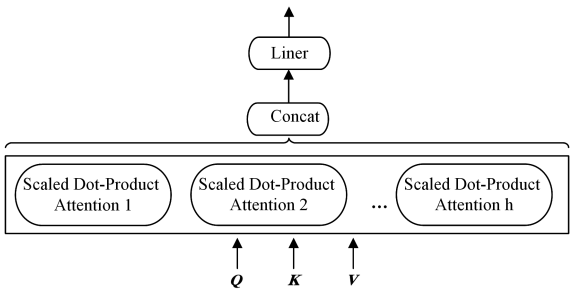


图 3 多头自注意力机制[11]

Fig. 3 Multi-head httention[11]

每个多头模块的计算过程由式(11)表示，式(12)表示将多个自注意力头的结果进行拼接后转换为特定维度的输出向量。

$$head_i = attention(\boldsymbol{QW}_i^{\boldsymbol{Q}}, \boldsymbol{KW}_i^{\boldsymbol{K}}, \boldsymbol{VW}_i^{\boldsymbol{V}}) \tag{11}$$

$$MHead(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = Concat(hea\,d_i, \cdots, hea\,d_h)W^0 \tag{12}$$

其中，$\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}$ 分别代表查询矩阵、键矩阵和值矩阵；$\boldsymbol{W}_i^{\boldsymbol{Q}}$，$\boldsymbol{W}_i^{\boldsymbol{K}}$，${}^{\boldsymbol{V}}\boldsymbol{W}_i$ 分别表示对 $\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}$ 进行变换的矩阵，$\boldsymbol{W}_i^{\boldsymbol{Q}}, \boldsymbol{W}_i^{\boldsymbol{K}}, \boldsymbol{W}_i^{\boldsymbol{V}} \in R^{d_{mod} \times d_k}$；$h$ 代表自注意力头数。$MHead(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})$ 代表由多头信息拼接变换后的多头注意模块的输出，其长距离特征捕获的能力受 Multi-Head 数量的影响，数量越多，特征捕获效果越好。由于其内部是一系列的矩阵乘法操作，所以并行化能力优于 CNN 和 RNN 结构。

3.2.3 Feed-Forward Networks

前馈神经网络(Feed Forward)前的残差模块(Residual Block)对多头自注意力层的输出与编码器的输入进行求和后再进行 Dropout(dropout 率为 0.1)操作来减少冗余信息。归化模块(Layer Normal)利用单个样本数据上的均值和标准差来不断调整神经网络的中间输出，从而使整个神经网络在各层的中间输出的数值更稳定，同时具有正则化的效果。Layer Normal 层计算的公式化表示如下：

$$m = \frac{1}{D}\sum_1^D x_i \tag{13}$$

$$\sigma = \sqrt{\frac{1}{D}\sum_1^D (X_i - m)^2} \tag{14}$$

$$LN(x) = \alpha \times \frac{(x-m)}{\sqrt{\alpha^2 + \varepsilon}} + \beta \tag{15}$$

其中，x_i 代表经自注意力层的输出与编码器的输入融合后的向量的第 i 维；m 表示输入 x 的均值，代表输入 x 的标准差；α 和 β 是可训练参数，ε 是为防止除数为 0 而设的小数。

经残差模块与归一化模块处理后的结果传递到前馈(feed-forward)神经网络中，其计算过程由式(16)表示。每个位置的单词对应的前馈神经网络都完全一样。

$$FFN(x) = \max(0, x w_1 + b_1)w_2 + b_2 \tag{16}$$

这里是两层网络，第一层采用 ReLU 激活函数来达到非线性变化的目的；后一层是线性函数，其中 w_1, w_2 为权值，b_1, b_2 为偏置。之后再次经过残差模块和归一化模块，一个 transformer_ block 计算完毕，我们的模型通过堆叠两个 transformer_block 来获得分层，动态的语义特征向量使得后续模块的处理更加高效。

3.3 多通道卷积神经网络模块

通过设置不同大小、数量的卷积核对经过上述模块特征融合后的动态词向量进行局部特征抽取与融合，来达到分类的目的。针对文本领域，局部特征可认为由若干单词(N-gram)组成的滑动窗口，卷积神经网络独特的结构可自动地对文本的 N-gram 特征进行组合和筛选，获得不同抽象层次的语义信息，可有效避免传统的 n-gram 方式可能会出现的随着 n 的增大参数空间呈指数增长和数据稀疏带来的需要复杂平滑机制的问题。

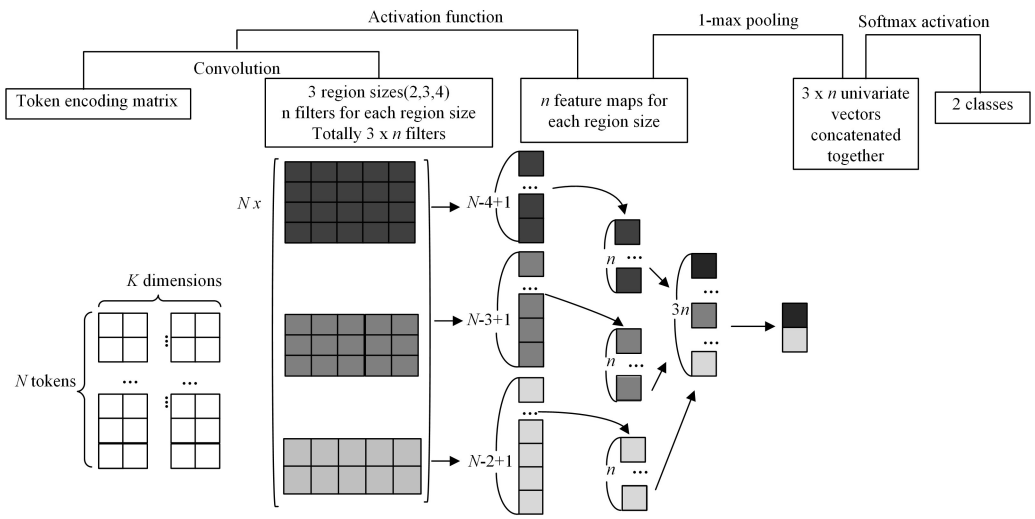


图 4 用于句子分类的 CNN 体系结构图示[18]

Fig. 4 Illustration of a CNN architecture for sentence classification[18]

具体处理思想如图 4 所示，将经过动态词向量表征模块(即两层 transformer_block)处理之后的融合向量(下文以 token 指代)按位置进行竖直拼接作为多通道卷积神经网络模块的输入。一个由 n 个 token 组成的句子可以表示为 $x_{1:n} = x_1 \oplus x_2 \oplus \cdots \oplus x_n$，其中⊕表示竖直拼接。将输入表示为 $n \times k$ 的矩阵，k 为词嵌入的维度大小，利用一维卷积的处理方式对其进行计算。首先使用一个 filter $\boldsymbol{W}(\boldsymbol{W} \in \boldsymbol{R}^{h \times k})$ 在输入矩阵上从上到下滑动，当滑动到某一位置时，filter $\boldsymbol{W}$ 内数组与所

覆盖的输入矩阵窗口 $x_{i:i+h-1}$ 内数组进行卷积操作，即逐元素相乘得到的结果再求和，进而产生一个特征 c_i，即 $c_i=f(w^* x_{i:i+h-1}+b)$，其中 $x_{i:i+h-1}$ 代表由输入矩阵的第 i 行到第 $i+h-1$ 行所组成的一个大小为 $h\times k$ 的窗口，具体由 $x_i, x_{i+1}, \cdots, x_{i+h-1}$ 拼接而成，h 表示窗口中的单词数，$\boldsymbol{W}$ 为 $h\times k$ 维的权重矩阵，b 为偏置参数，f 为非线性函数。最终 $C=[c_1, c_2, \cdots, c_{n-h+1}]$（$C\in R^{n-h+1\times 1}$）是我们获得的 feature map。每一次卷积操作相当于一次特征向量的提取，通过定义不同的窗口，就可以提取出不同的特征向量，构成卷积层的输出。我们采用 $h=(3,4,5)$ 的不同 size 的 filter，每种 size 的 filter 个数为 100，对句子矩阵进行卷积并生成可变长度大小为 $(n-h+1)\times 1$ 的特征图（filter 的大小和句子长度决定了特征图的维度），在每个特征图上执行 1-max 池化，即为从每个滑动窗口产生的特征向量中筛选出一个最大的特征并将这些特征拼接起来得到一个定长的向量表示，然后通过 softmax 激活函数达成分类的目的，同时在 softmax 层使用 dropout 和 L2 正则化来降低模型过拟合的概率。

4　对比实验及参数设置

4.1　数据集

我们分别在电影评论数据集 IMDb 和 SST-2 上测试本文方法的有效性。IMDb 数据集共包含 50 000 条来自美国电影评价网站的数据，文本平均长度为 292，按照情感极性可以划分为积极（Pos）和消极（Neg）两种情感类别。SST-2 数据集来自于 Stanford 情感树库，约有 11 855 条文本影评，同样也包含正（Pos）、负（Neg）两种情感类别。本文所用的实验数据的统计如表 1 所列。

表 1　实验中使用的数据集

Table 1　Dataset used in experiment

Data set	Category	Average length	Training/validation/testing number
IMDb	2	292	21 143/3 857/25 000
SST-2	2	18	8 544/1 101/2 210

4.2　参数设置

在本文实验中的各项超参是我们通过实验对 SST-2 的数据集进行网格搜索所获得的，其中 transformer_block 层数和自注意力机制头数是我们自己拟定的。模型的具体超参数设置如表 2 所列。

表 2　超参数设置

Table 2　Hyperparameter settings

Hyperparameter	Settings
基础词向量	glove
词嵌入维度	300
卷积核大小	(3,4,5)
每个 size 下的 filter 个数	100
激活函数	ReLU
池化策略	1-max poolong
优化器	Adadelta
Bath Size	50
Dropout 率	0.5
L2 正则化系数	0.001
transformer_block	2 层
自注意力机制头数	6

4.3　实验对比

对本文所提模型与对照实验进行简介。

N-gram 词袋模型＋支持向量机（Support Vector Machine，SVM）：以 n 取（1，2，3）的 unigrams，bigrams 和 trigrams 分别对应字，词级别的特征，利用矩阵表示数据集其中每行表示一条文本，每列使用 TF-IDF（term frequency-inverse document frequency）分数来填充用以表示提取的特征。将经典的字/词/字词融合级词袋模型结合线性核的 SVM 传统机器学习模型作为我们的 baseline。

我们通过 3 种不同特征提取器 CNN，RNN，Transformer 分别设置了 3 组实验，其中 rand 表示使用随机初始化的（300 维）词向量作为单词的表示向量，static 表示使用 glove 预训练词向量作为单词的表示。

DCNN[13]：引入宽卷积和动态 K-max 池化的思想来保证句子边缘信息不丢失，同时使得不同长度句子提取的语义特征数相同。

rand-MCNN：通过多通道卷积网络（MCNN）进行局部特征的提取及融合。

static-MCNN：保持词向量在训练过程中不变，同时调节模型其他部分的参数。特征提取器同上。

Tree-LSTM[14]：相比于传统 LSTM 增加多个遗忘门，进而可以有选择性地从子节点获取信息。

rand-biLSTM：堆叠两层双向 LSTM 进行语义特征提取。

static-biLSTM：保持词向量在训练过程中不变，同时调节模型其他部分的参数。

Rand-tf-MCNN：通过两层 teansformer_block 进行特征提取，在训练过程中动态调整分布式表示的词向量，之后通过 MCNN 进行特征融合及降维。

static-tf-MCNN：在词向量到卷积分类器间堆叠两层 transformet_block 特征提取器，在训练过程中动态调整分布式表示的词向量。使用 transformer_block 最高层特征作为卷积分类器的输入。

5　结果分析

5.1　结果分析

针对特定任务，应用预训练模型学习的通用语言表示向量时有两种主要方式，其一是特征提取（Feature Extraction），保持词嵌入层参数不动，使用预训练表示作为额外的特征，并针对不同任务设计特定结构去处理下游 NLP 任务（例如 ELMO）；另外一种是微调（Fine-Tuning），词嵌入层参数针对特定任务使用相对应的训练集随着训练过程进行更新（例如 GPT，BERT）。

表 3　模型在不同数据集上的准确度统计

Table 3　Accuracy statistics of model on different data sets

（单位：%）

Feature Extractor	model	IMDb	SST-2
Bow＋SVM	Word ngram	86.7	81.4
	Char n-gram	87.3	83.2
	(Word＋Char)-ngram	**88.5**	**84.6**
CNN	DCNN[13]	86.8	86.8
	rand-MCNN	86.2	87.2
	static-MCNN	**88.4**	**88.1**
RNN	Tree-LSTM[14]	**88.4**	**88.00**
	random-biLSTM	82.8	85.6
	static-biLSTM	86.6	87.60
transformer	random-tf MCNN	88.1	88.5
	static-tf MCNN	**90.4**	**90.20**

我们选用 transformer 作为语义特征提取器来获得特定任务和数据集的动态词向量表征，因此我们所有 transformer 模块下的实验均使用 Fine-Tuning 的方式来进行模型训练，各组实验结果统计如表 3 所列，后续将对组内及不同组间结果进行对比分析。

5.2 BOW+SVM VS 三大特征提取器

如图 5 所示，Bow+SVM 在 IMDb 数据集上的最优模型(Word+Char)-ngram 的准确率为 88.5%弱于 transformer 特征提取器中的最优模型 static-tf-MCNN 的准确率 90.4%，高于 CNN 和 RNN 特征提取器的最优模型 static-MCNN 和 Tree-LSTM 的 88.4%。如图 6 所示，在 SST-2 数据集上最优模型的准确率均低于三大特征提取器中的最优模型的准确率。特征工程+机器学习分类器的方法普遍弱于深度学习方法，其原因是词袋模型没有考虑词序，对句子语义理解存在较大偏差，其中 TF-IDF 主观臆断程度大，认为词频和逆文档频率针对文本分类任务是重要因素，但在情感二分类任务上，文本普遍较短，逆文档意义不大。IMDb 数据集的平均长度为 292，SST-2 数据集的平均长度为 18，由结果推论词袋模型在多分类、长文本领域可能效果会更好。

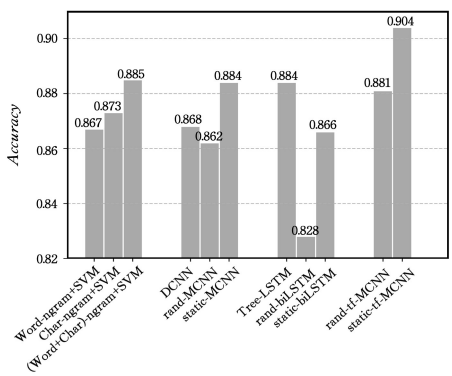


图 5 不同模型在 IMDb 数据集上的准确率对比

Fig. 5 Comparison of accuracy of different models on IMDb data set

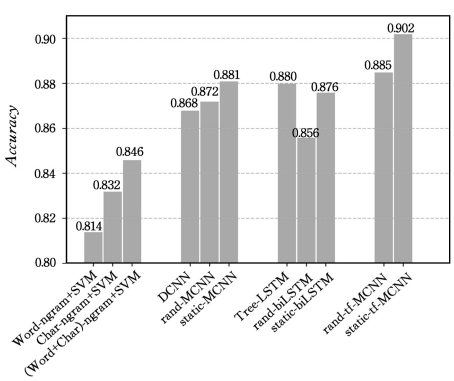


图 6 不同模型在 SST-2 数据集上的准确率对比

Fig. 6 Comparison of accuracy of different models on SST-2 data set

5.3 DCNN vs MCNN

DCNN 通过在句子边缘补空值再利用宽卷积(Wide Convolution)提取句子特征，可有效提取句子边缘信息，并且池化时用动态 K 最大池化(dynamic k-max pooling)替换传统的 K 最大池化(k-max pooling)或平均池化(mean-pooling)来保留特征原有的顺序，达到利用动态卷积神经网络的方式对句子进行建模的目的。如图 7 所示，其在 IMDb 数据集上准确率达 86.8%，高于 rand-MCNN 方法的 86.2%。其中 static-MCNN 模型在两个数据集上均达到最优效果，其通过多通道卷积进行局部 ngram 特征提取并且池化时采用 1-max pooling 有效进行特征融合及降维，使得句子语义特征的提取效果最好，在 IMDb 和 SST-2 测试集上准确率分别为 88.4%和 88.1%。而 rand-MCNN 模型的准确率分别仅为 86.2%和 87.2%，与最优模型准确率相差 2.2%和 0.9%，表明预训练通用的词向量对情感分类任务的效果有较大提升。DCNN 通过精心设计特征提取及融合的方法，使得情感语义特征的表达更完整。其采用本文的测试集准确率均为 86.8%，虽未超过 static-MCNN 模型，但在其他任务或数据集上效果可能会更优。

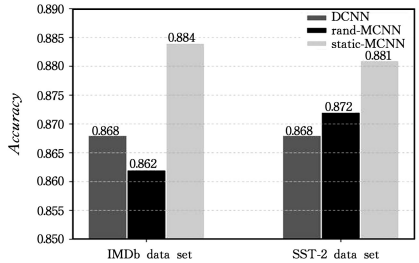


图 7 基于 CNN 提取器的不同模型准确率对比

Fig. 7 Comparison of accuracy of different models based on CNN extractor

5.4 Tree-LSTM vs biLSTM

LSTM 可用于处理序列建模任务，其通过引入门控机制有效解决了 RNN 处理序列信息时存在的梯度消失和无法有效捕获长距离依赖的问题，其双向结构更能完整地提取句子的语义信息，如图 8 所示，其在 IMDb 和 SST-2 数据集上，通过堆叠两层 biLSTM 可以达到 86.6%和 87.6%的准确率。

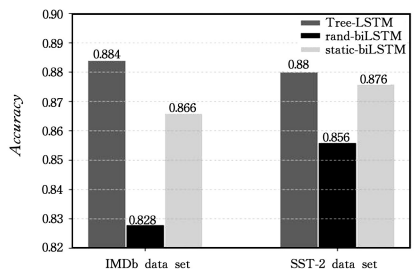


图 8 基于 RNN 提取器的不同模型准确率对比

Fig. 8 Comparison of accuracy of different models based on RNN extractor

Tree LSTM 在 LSTM 的处理线性顺序链的基础上，提出通过树形拓扑结构将单词和短语组合在一起并引入多个遗忘门来分别对应每个子单元，使得 Tree-LSTM 有选择性地从子节点获取更加丰富的信息，在 IMDb 和 SST-2 数据集上均取得了最优效果 88.4%和 88%。相比于 static-biLSTM 模型，其准确率分别提升了 1.8%和 0.04%，证明了设计更复杂和优秀的特征提取结构的必要性。随机初始化的词向量 rand-biLSTM 模型和静态词向量 static-biLSTM 模型在其他参数一致的情况下，准确率在 IMDb 和 SST-2 数据集分别相差 3.8%和 2%，差距较大，再次体现了预训练词向量的重要性。

5.5 三大特征提取器

深度学习在各个 NLP 任务中屡创佳绩，其功劳当属自然

语言领域的特征提取结构。三大特征提取器自动学习抽取对于特定任务的最优特征，实现了深度学习领域的“端到端”模型。CNN 结构应用于 NLP 领域类似于 n-gram 模型，通过不同大小的卷积核对文本提取不同层次的语义信息，之后通过池化操作保留特征信息中的重要信息，同时达到降维的作用，如图 9、图 10 所示，其在 IMDb 和 SST-2 数据集上的最优准确率分别为 88.4%和 88.1%。RNN 结构通过设置隐藏层状态使得处理当前时刻的输入时会利用上一时刻的输出。LSTM 结构通过引入门控机制来解决长距离依赖及梯度消失问题，如图 9、图 10 所示，其在 IMDb 和 SST-2 数据集上的最优准确率分别为 88.4%和 88%。transformer 结构通过多头注意力机制(Multi-head attention)来进行不同位置的语义信息融合，并利用残差模块来降低冗余信息，同时使用 Layer normal 降低过拟合，加快模型训练，最终借助前馈神经网络(Feed Forward)达到语义的分层表示，如图 9、图 10 所示，其在 IMDb 和 SST-2 数据集上的最优准确率分别为 90.4%和 90.2%。通过不同数据集上的实验结果发现，transformer 特征提取器的效果均优于 CNN 和 RNN 及在其基础上的改进模型的效果，这表明 transformer 的语义特征提取能力更优。

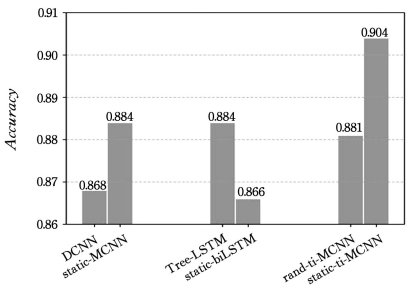


图 9　IMDb 数据集上不同模型准确率对比

Fig. 9　Comparison of accuracy of different models on IMDb data set

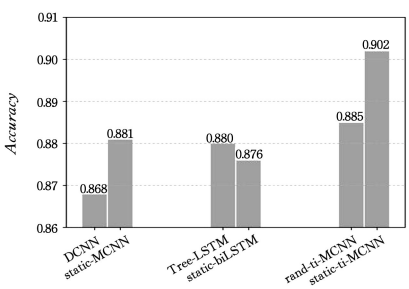
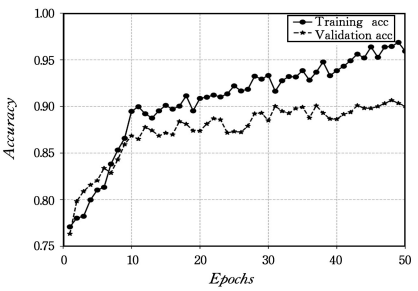


图 10　SST-2 数据集上不同模型准确率对比

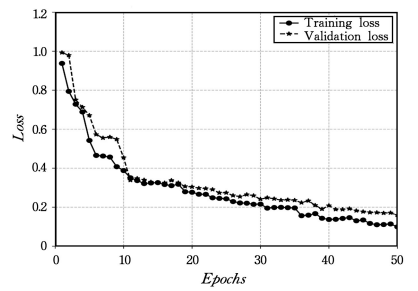
Fig. 10　Comparison of accuracy of different models on SST-2 data set

5.6　模型训练过程

将本文所提模型分别在 IMDb 和 SST-2 数据集上进行实验。图 11 和图 12 给出了两数据集上训练集和验证集的准确率及 loss 值的变化情况。两数据集中训练集的准确率在前 10 个 epoch 随着训练次数增加而稳步较大幅增大，在后 40 个 epoch，准确率随训练次数增加而逐步提升。其中 IMDb 数据集当进行到第 10 个 epoch 时，训练集准确率达到 90%左右之后小幅度震荡并稳步上升，最终可达到 96%左右，验证集准确率在第 12 个 epoch 时达到 87.3%，之后逐渐增加最终稳定在 90%左右，表明模型即便增加 L2 正则化之后也有轻微的过拟合现象出现，训练集和验证集的 *loss* 值均逐渐下降，最终训练集 *loss* 值为 0.104，验证集 *loss* 值为 0.162。SST-2 数据集当进行到第 9 个 epoch 时，训练集准确率达 90%，之后经过 2 个 epoch 达到 94.2%并逐渐稳定，最终达到 96.5%左右，验证集准确率在第 22 个 epoch 达到 90%后，逐渐稳定，同样存在轻微过拟合现象，训练集与验证集 *loss* 值同步下降，最终训练集 *loss* 值为 0.108，验证集 *loss* 值为 0.153。本文模型最终在 IMDb 和 SST-2 测试集上准确率分别达到 90.4%和 90.2%。



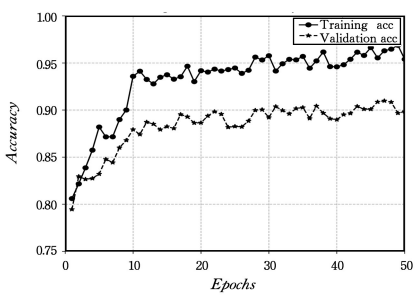
(a) Training and validation accuracy on IMDb



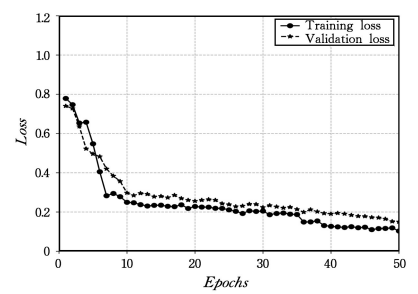
(b) Training and validation loss on IMDb

图 11　MCNN-Transformer 在 IMDb 数据集的训练过程

Fig. 11　MCNN-Transformer process on IMDb data set



(a) Training and validation accuracy on SST-2



(b) Training and validation loss on SST-2

图 12　MCNN-Transformer 在 SST-2 数据集训练过程

Fig. 12　MCNN-Transformer process on SST-2 data set

结束语　本文通过设置一系列的对比实验进行分析，发现：1)预训练词向量的通用表示针对自然语言处理领域的文本情感分析任务有明显提升效果，同时针对特定数据集的动态语义表征效果会更优；2)深度学习的“端到端”模型的建立得益于三大特征提取器的强大语义特征的提取及融合的能

力;3)针对特定任务和数据集的复杂模型能有效提高准确率。在短文本情感分类领域,更有效的语言特征表示及更充分的语义特征抽取模型的提出与改进将始终是关注的热点。

参 考 文 献

[1] HU M Q, LIU B. Mining and Summarizing Customer Reviews [C]// Proc of the 10th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2004: 168-177.

[2] PANG B, LEE L, VAITHYANATHAN S. Thumbs up? Sentiment Classification using Machine Learning Techniques[C]// Proc of Empirical methods in Natural Language Processing. Cambridge, MA: MIT Press, 2002: 79-86.

[3] SALEH M R, MART N-VAILDIVIA M T, MONTEJO-R E, et al. Experiments with SVM to classify opinions in different domains[J]. Expert Systems with Applications, 2011, 38(12): 14799-14804.

[4] KIM Y. Convolutional neural networks for sentence classification[C]// Proceedings of 2014 Conference on Empinical Methods in Natural Language Processing. Daha, Qatar, 2014: 1746-1751.

[5] ZHU X D, SOBIHANI P. Long short-term menory over recursive structures [C]// Proc. of Int. Conf. on Machine Learning. New York: ACM, 2015: 1604-1612.

[6] YUAN H J, ZHANG X, NIU W H, et al. Research on text sentiment analysis of multi-channel convolution and two-way GRU model with attention mechanism[J]. Journal of Chinese Information Processing, 2019, 33(10): 109-118.

[7] ZHAO Y O, ZHANG J Z, LI Y B, et al. Sentiment analysis combining word embedding based on language model and multi-scale convolutional neural network[J]. Journal of Computer Applications, 2020, 40(3): 651-657.

[8] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality [C]// NeurIPS. 2013.

[9] PENNINGTON J, SOCHER R, MANNING C D. GloVe: Global vectors for word representation[C]// EMNLP. 2014.

[10] PETERS M E, NEUMAN N, et al. Deep contextualized word representations [C]// Proceedings of North American Chapter of the Association for Computational Linguistics: ACL, 2018: 1-9.

[11] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need [C]// Proceedingsof Annual Conference on Neural Information Processing Systems. Long Beach, USA, 2017: 1-5.

[12] YOSINSKI J, CLUNE J, BENGIO Y, et al. How transferable are features in deep neural networks? [C]// Advances in Neural Information Processing Systems. 2017: 6000-6010.

[13] QIAN Q, TIAN B, HUANG M, et al. Learning tag embeddings and tag-specific composition functions in recursive neural network[C]// Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. Stroudsburg: Association for Computational Linguistics, 2015: 1365-1374.

[14] TAI K S, SOCHER R, MANNING C D. Improved semantic representations from tree-structured long short term memory networks [C]// Proceedings of Annual Meeting of the Association for Computational Linguistics. ACL, 2015: 1556-1566.

[15] RADFORD A, NARASIMHAN K, SALIMANS T, et al. Improving language understanding by generative pre-training[OL]. https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/language-unsupervised/language_ understanding _ paper. pdf.

[16] DEVLIN J, CHANG M W, LEE K, et al. BERT: pre-training of deep bidirectional transformers for language understanding [C]// NAACL-HLT. 2019.

[17] ZOPH B, GHIASI G, et al. 2020. Rethinking Pre-training and Self-training[OL]. https://arxiv. org/abs/2006. 06882.

[18] ZHANG Y, WALLACE B. A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification[OL]. https://arxiv. org/abs/1510. 03820.

HUO Shuai, born in 1994, postgraduate. His main research interests include emotion analysis and deep learning.

PANG Chun-jiang, born in 1965, associate professor. His main research interests include artificial intelligence and internet of things.