Advances in Applied Mathematics 应用数学进展, 2022, 11(10), 7012-7025
Published Online October 2022 in Hans. http://www.hanspub.org/journal/aam
https://doi.org/10.12677/aam.2022.1110744

Hans汉斯

结合自注意力和归一化的MAC_BiLSTM文本分类模型

原明君，江开忠，杨　洋，惠岚昕

上海工程技术大学，数理与统计学院，上海

收稿日期：2022年9月12日；录用日期：2022年10月2日；发布日期：2022年10月12日

摘　要

针对文本分类任务中关键特征分布不均匀和双向长短期记忆网络(BiLSTM)局部特征信息提取不足的问题，提出了一种基于自注意力机制(Self_Attention)和归一化的多通道MAC_BiLSTM文本分类模型。在双向长短期记忆网络层之后加入自注意力机制并进行层归一化，同时将BiLSTM通道的信息与最初的词向量信息融合，输入卷积通道，再分别采用自注意力赋予词卷积方式重新计算后信息的词权重，并进行批归一化，重复两次之后再进行池化，最终将CNN通道池化后的特征信息与BiLSTM通道信息进行特征融合，并通过Softmax分类器得出分类结论。在模型的设计环境中，模型使用了更加平滑的Mish激活函数代替Relu，通过和其他深度学习模型在多个数据集上的比较，结果表明，所提出的模型与其他模型相比具有更好的分类效果。

关键词

自注意力机制，卷积神经网络，双向长短期记忆网络，归一化，多通道

MAC_BiLSTM Text Classification Model Based on Self-Attention and Normalization

Mingjun Yuan, Kaizhong Jiang, Yang Yang, Lanxin Hui

School of Mathematics, Physics and Statistics, Shanghai University of Engineering Science, Shanghai

Received: Sep. 12th, 2022; accepted: Oct. 2nd, 2022; published: Oct. 12th, 2022

Abstract

To address the problem of uneven distribution of key features in traditional text classification tasks and insufficient local feature extraction ability of Bi-directional Long Short-Term Memory

文章引用: 原明君, 江开忠, 杨洋, 惠岚昕. 结合自注意力和归一化的 MAC_BiLSTM 文本分类模型[J]. 应用数学进展, 2022, 11(10): 7012-7025. DOI: 10.12677/aam.2022.1110744

(BiLSTM), a multi-channel text classification model based on self-attention mechanism and normalization is proposed. Self-attention mechanism and layer normalization are added after BiLSTM layer. The BiLSTM channel information and the initial word vector information are fused to input into the convolution channel. The weight of information after the convolution operation is given by self-attention, and the batch normalization is carried out. After repeated twice, the pooling is carried out. Then the feature information after max pooling and BiLSTM channel information are fused to input Softmax, and obtain the classification results. In the process of model operation, the proposed model uses smoother Mish activation function instead of Relu. Through comparison experiments with other deep learning models on multiple datasets, the results show that the proposed model has better classification accuracy than other models and has better classification performance.

Keywords

Self-Attention Mechanism, Convolutional Neural Network, Bi-Directional Long Short-Term Memory Network, Normalization, Multi-Channel

1. 引言

文本分类在舆情研究、问答系统、客户的信息服务等诸多方面起到了巨大的作用，怎样对文本进行合理的分类，成为当前自然语言研究方面探讨的焦点课题。不少国内外专家已经提供了许多模型来处理这一现象，包括循环神经网络(Recurrent Neural Network, RNN) [1]、长短期记忆网络(Long Short-Term Memory, LSTM) [2]、卷积神经网络(Convolutional Neural Networks, CNN) [3]、自注意力机制模型[4]等。

本文模型基于 BiLSTM 层使用注意力机制在输出数据中获得注意力值，进而对层进行归一化；同时把 BiLSTM 通道的结果和最初的词向量数据结合，在两个卷积层中根据自注意力赋予词卷积运算后结果的权重，分别进行批归一化，重复两次之后分别进行池化，最终将 CNN 通道池化后的特征信息与 BiLSTM 通道信息进行融合，结果表明本文提出的模型具有较好的分类效果。

2. 相关工作

Bengio 等[5]构建出神经网络语言模型(NNLM)，为神经网络在 NLP 的发展指明思路。为获得更高效的词向量表示，Mikolov 等[6]提出两种词向量模型，CBOW (Continuous Bag-of-Words)和 Skip-Gram。由于 CNN 能够准确地提取文本的局部特征，RNN 能够有效地处理上下文数据。Zhang 等[7]将 LSTM 和 CNN 相融合，最终验证了 RNN 与 CNN 进行组合的有效性。吴汉瑜等[8]和梁顺攀等[9]都主张将 BiLSTM 和 CNN 融合，利用双向传输机制获得文本完整的上下文数据，最终得到特征融合可以有效提高文本分类的准确性。张小川等[10]将 BiGRU 与 CNN 结合，采用 CNN 获取词向量的局部表示，利用 BiGRU 获取全局上下文表示，结果表明所提模型具有良好的文本建模能力。

自注意力机制能够对文本的重要特征聚焦，所以，有些研究人员将以上三个模型组合起来，运用各自优点来获得更丰富的文本信息。陶志勇等[11]将自注意力机制引入 BiLSTM，用于短文本分析；蒲相忠等[12]将自注意力机制引入卷积神经网络；邓朝阳等[13]在门控循环单元的基础上设置注意力门，加强特征信息交互，以上模型均得到了较好的分类结果。陈农田等[14]提出一种多通道 CNN_BiGRU_att 中文文

本分类模型，模型使用 CNN 获取局部特性，使用 BiGRU 获取上下文的全局信息；陈可嘉等[15]提供了一个基于自注意机制和多通道 CNN 的 SAttBiGRU_MCNN 文本分析模型，在仿真实验结果中取得了良好成效，提高了文本分类的准确性。

3. MAC_BiLSTM 文本分类模型的构建

3.1. 整体模型结构

MAC_BiLSTM 文本分类模型主要由三个部分组成：第一个部分是由 BiLSTM、自注意力机制以及层归一化构成的 BiLSTM 通道；第二个部分是由通过 BiLSTM 和自注意力机制后的特征向量，与最初词向量嵌入层融合而成的特征向量组成；第三个部分是由 CNN 拼接而成的并行通道组成，该通道输入向量为第二个部分输出结果。其中，BiLSTM 通道主要用于获取文本信息中的长距离依赖关系，再将 BiLSTM 通道输出结果经过 CNN 通道可进一步对文本的局部特征信息进行提取，从而获得更加准确的文本信息。模型具体结构如图 1 所示。

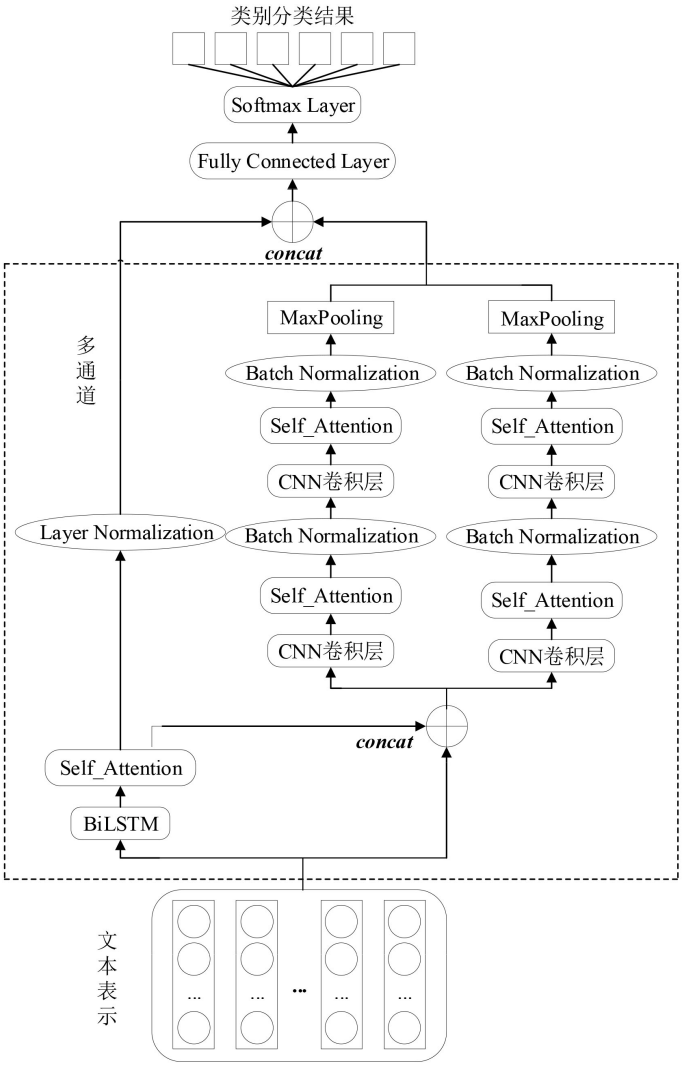

Figure 1. MAC_BiLSTM text classification model
图 1. MAC_BiLSTM 文本分类模型

3.2. 文本表示

Word2vec 目前主要有两种模型来学习单词的分布式表示，一种是使用中间词来计算该单词上下文的 Skip-Gram，另一个是通过上下文数据来估计中间单词的 CBOW。本文训练词向量主要使用 Skip-Gram 模型，其网络结构图如图 2 所示。

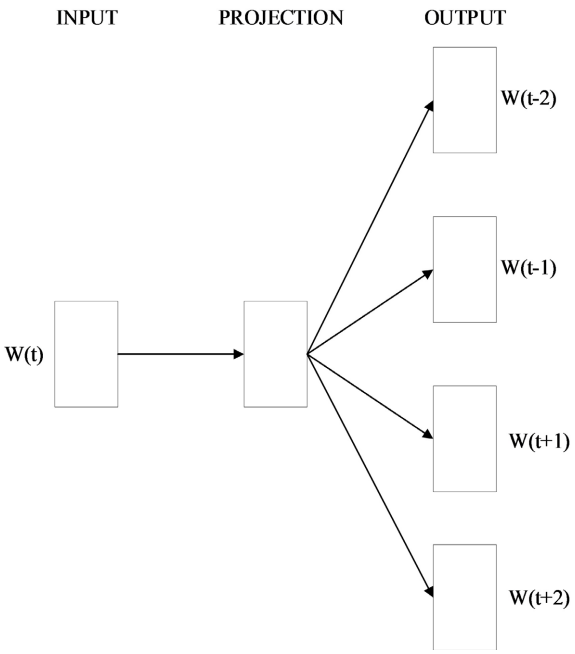

Figure 2. Skip-Gram structure diagram
图 2. Skip-Gram 结构示意图

3.3. BiLSTM 通道

3.3.1. BiLSTM + Attention

LSTM 模型 t 时刻的计算过程如下：

$$f_t = \sigma\left(W_f \cdot \left[h_{t-1}, x_t\right] + b_f\right) \tag{1}$$

$$i_t = \sigma\left(W_i \cdot \left[h_{t-1}, x_t\right] + b_i\right) \tag{2}$$

$$\tilde{C}_t = \tanh\left(W_C \cdot \left[h_{t-1}, x_t\right] + b_C\right) \tag{3}$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{4}$$

$$o_t = \sigma\left(W_o \cdot \left[h_{t-1}, x_t\right] + b_o\right) \tag{5}$$

$$h_t = o_t \cdot \tanh\left(C_t\right) \tag{6}$$

在运算过程中，σ 表示为激活函数，f_t 表示 t 时刻遗忘门的节点操作，o_t 表示输出门 t 时刻输出，W_* 为权值，b_* 为偏置项，x_t 为时间 t 的输入向量，h_{t-1} 为前一步计算的输出状态，h_t 为最终状态的输出，C_{t-1} 为上一时刻的单元状态，C_t 为当前单元状态。

BiLSTM 模型在处理文本时可学到大量双向语义数据，可对文本内容进行更好地分类，其结构如图 3 所示。

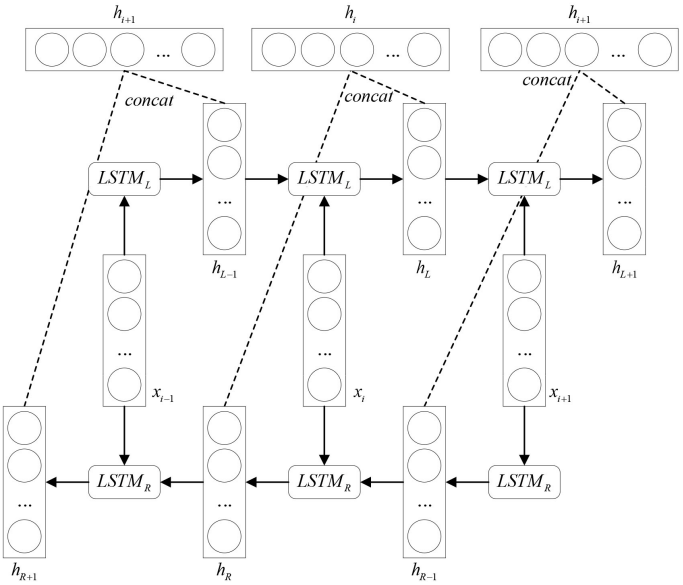

Figure 3. BiLSTM model diagram
图 3. BiLSTM 模型图

如图，BiLSTM 由正向 LSTM 和逆向 LSTM 构成，x_{i-1}、x_i 和 x_{i+1} 代表 $i-1$、i 和 $i+1$ 时刻的输入信息；正向隐藏层 LSTM_L 的输出为 h_L，逆向隐藏层 LSTM_R 的输出为 h_R，*concat* 表示不同隐藏层输出向量的拼接，BiLSTM 隐藏层的输出 h_t 表示为：

$$h_t = \left[h_L, h_R\right] \tag{7}$$

自注意力机制可以在保留原文特点的基础上突出文章重要特点，把注意力集中到某些对文章比较重要的词语上，注意力机制的计算流程如下：

$$u_t = \tanh\left(W_w h_t + b_w\right) \tag{8}$$

$$\alpha_t = \text{Softmax}\left(u_t^{\text{T}}, u_w\right) \tag{9}$$

$$s_1 = \sum_t \alpha_t h_t \tag{10}$$

其中，u_t 为 h_t 的自注意力隐藏层表示，W_w 为权重矩阵，b_w 为偏置单元，α_t 为 u_t 经过 Softmax 层后获得的归一化权重。最后将 BiLSTM 输出值 h_t 与权重值 α_t 进行点乘并求和，得到最终输出值 s_1。

3.3.2. Layer Normalization

Hinton 等[16]提出适用于 RNN 模型的层归一化(Layer Normalization)加快模型学习速度，具体根据以下公式将网络某一层的所有神经元输入进行归一化：

$$\mu^l = \frac{1}{H}\sum_{i=1}^{H} a_i^l \tag{11}$$

$$\sigma^l = \sqrt{\frac{1}{H}\sum_{i=1}^{H}\left(a_i^l - \mu^l\right)^2} \tag{12}$$

$$\hat{a}^l = \frac{a^l - \mu^l}{\sigma^l} \tag{13}$$

$$y = g \cdot \hat{a}^l + b \tag{14}$$

其中，μ^l、σ^l 分别代表各层神经元的均值和方差，H 代表网络各层节点个数，a_i^l 代表第 l 隐藏层输出，y 代表模型标准化值，g 和 b 分别表示向量的缩放和平移。

3.4. CNN 通道

3.4.1. CNN + Attention

为了提取更加丰富的局部特征，本文使用多层 CNN。该通道输入为经过词嵌入层映射得到向量与经过 BiLSTM + Attention 通道之后词向量的融合。假设经过 *concat* 后的矩阵为 S_0，模型采用 2 个并行的 CNN 通道对 S_0 进行局部特征提取操作。为了进一步提高双通道 CNN 的特征提取能力，获取文本的多元特征，在 2 个 CNN 通道再加入一层卷积对 CNN 进行优化，加强文本局部特征表达能力，同时引入自注意力机制层与批归一化层进一步加强模型的学习能力。

设输入词向量为 $S = [x_1, x_2, \cdots, x_N]$，其中，$x_i \in R^n$，$x_i$ 代表输入文本中第 i 个词所对应的词向量，n 代表选定的词向量维度。将卷积核定义为 W，则卷积层的运算过程可以表示为：

$$c_i = f(W \otimes S + b) \tag{15}$$

其中，c_i 表示通过卷积计算输出的第 i 个特征值，$f(\cdot)$ 表示非线性激活函数，$\otimes$ 表示卷积运算，b 为偏置单元。

词向量矩阵 S 经过卷积运算，得到一个 CNN 卷积后的输出矩阵 $C_1 = [c_1, c_2, \cdots, c_N]$，$N$ 为词向量个数。接着将 C_1 输入自注意力机制层进行处理，计算过程如下：

$$u_i = \tanh(W_w c_i + b_w) \tag{16}$$

$$\alpha_i = \text{Softmax}(u_i^{\mathrm{T}}, u_w) \tag{17}$$

$$s_2 = \sum_i \alpha_i c_i \tag{18}$$

3.4.2. Batch Normalization

为增强模型的自适应能力和表达能力，加入 Batch Normalization 批标准化层[17]对通过 CNN 卷积层和自注意力机制层产生的输出向量加以处理。假定 Batch_size 为 m，Batch Normalization 层输入向量为 $B = [x_1, x_2, \cdots, x_m]$，则 Batch Normalization 计算流程如下：

$$\mu_B = \frac{1}{m}\sum_{i=1}^{m} x_i \tag{19}$$

$$\sigma_B^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_B)^2 \tag{20}$$

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}} \tag{21}$$

$$y_i = \gamma \hat{x}_i + \beta \tag{22}$$

其中，μ_B、σ_B 分别代表神经元各批次的均值和标准差，通过公式(22)，当前 Batch_size 第 i 个输入节点的值 x_i 变为均值为 0、方差为 1 的正态分布 $\hat{x}_i$，ε 为避免除零输入的极小值，γ 和 β 为缩放和平移参数。

最后，再将经 Batch Normalization 后的特征向量进行池化，计算流程如下：

$$\tilde{C} = \max(C) \tag{23}$$

3.5. Softmax 层

由于 CNN 通道中的池化运算将特征向量进行了降维，使 CNN 通道与 BiLSTM 通道输出向量呈现不同维度，因此需要对各通道输出矩阵进行处理，然后将经过 *concat* 函数融合后的向量存于 output，进行全连接运算。

本文采用 Softmax 分类器实现 output 的文本分类，得到各类别输出向量的概率分布，计算过程如下：

$$P\left(y^{(i)}=j \mid x^{(i)};\theta\right)=\frac{\exp\left(\theta_j^{\mathrm{T}}x^{(i)}\right)}{\sum_{n=1}^{k}\exp\left(\theta_n^{\mathrm{T}}x^{(i)}\right)} \tag{24}$$

其中，P 表示输入向量 x 分类到类别 j 的概率，θ 为模型训练的参数。

3.6. 模型激活函数

本文采用由 Diganta [18]提出的 Mish 激活函数。传统大多使用 Sigmoid、Tanh、Relu 激活函数，各函数公式为：

Sigmoid：
$$\sigma(z)=\frac{1}{1+\mathrm{e}^{-z}} \tag{25}$$

Tanh：
$$\tanh(z)=\frac{\mathrm{e}^{z}-\mathrm{e}^{-z}}{\mathrm{e}^{z}+\mathrm{e}^{-z}} \tag{26}$$

Relu：
$$f(z)=\begin{cases} z & z>0 \\ 0 & z\le 0 \end{cases} \tag{27}$$

Mish：
$$f(z)=z\cdot\tanh\left(\ln\left(1+\mathrm{e}^{z}\right)\right) \tag{28}$$

各激活函数对应的函数图像如图 4：

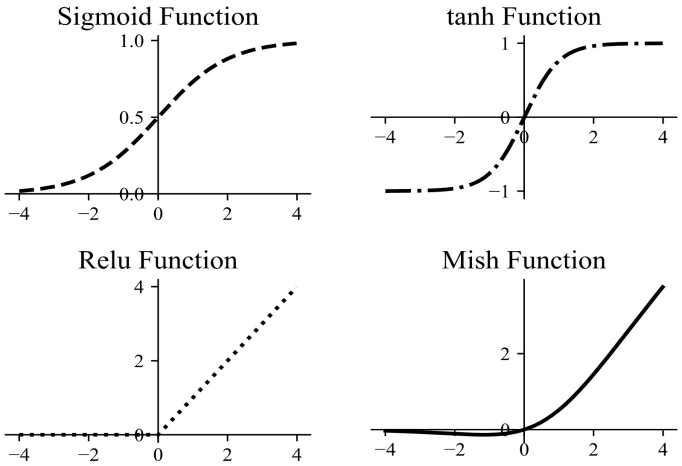

Figure 4. Activation function
图 4. 激活函数图像

对比 Sigmoid、Tanh、Relu 激活函数，Mish 函数正值能够到达任何高度，从而有更好的梯度流信息，而不像 Sigmoid 函数无负值流入，Tanh 容许进入过大的负值，以及 Relu 函数的硬零界限，能够更好地保障特征信息的流动。

4. 实验结果与分析

4.1. 实验环境与数据概述

本文的实验环境如表 1 所示：

Table 1. Setup of experimental environment
表 1. 实验环境设置

实验环境	详细信息
操作系统	Windows10
CPU	Intel(R) Xeon(R) CPU E5-2680 v4 @ 2.40 GHz
内存	28 G
显卡	NVIDIA GeForce RTX 3060
开发语言	Python 3.8.8

本文实验选取网络上公开的用于文本分类的四个中文数据集：THUNews 新闻数据集[19]、今日头条新闻数据集、online_shopping_10_cats (简称 os10c)数据集和 ChnSentiCorp (简称 Chn)数据集。

THUNews 新闻数据集由清华大学提供并公开，共涵盖经济、地产、股票、教育、科学、社会、政治、运动、游戏、文娱 10 类资讯。今日头条数据集涵盖 15 种资讯，实验提取其中 8 类：科技、娱乐、体育、军事、金融、汽车、文娱、教育。os10c 数据集收录了图书、平板、手机、果蔬、洗发水、热水器、蒙牛、服装、电脑、酒店 10 类评论信息。Chn 数据集由哈工大谭松波教授整理并提供，包含酒店、笔记本、书籍三个领域正向、负向 2 类情感数据。

本文将数据集随机分为训练集、测试集和验证集，数据集概况如表 2 所示：

Table 2. Dataset statistics
表 2. 数据集统计表

数据集	总样本	训练集	验证集	测试集	类目
THUNews	300,000	260,000	20,000	20,000	10
今日头条	90,000	50,000	20,000	20,000	8
os10c	60,000	40,000	10,000	10,000	10
Chn	9000	5000	2000	2000	2

4.2. 数据预处理与模型参数设置

本文利用 Jieba 对输入数据集进行分词处理，利用 Word2vec 模型中的 Skip-Gram 方法开展词向量训练，得到的词向量分布式表示维度为：50、100、150、200、300。同时引入数据增强(Data Augmentation)技术，使有限数据集获得更大数据量，改善模型稳健性，增强模型泛化能力。

模型的具体参数设置如表 3 所示。

4.3. 评价指标

大多数分类模型的评判标准是：准确率(Accuracy)、精确率(Precision)、F1 值(F-measure)及召回率(Recall)。相关的混淆矩阵结构图如表 4 所示。

Table 3. Setup of model parameters
表 3. 模型参数设置

参数名称	设置值
卷积核尺寸	3
Adam 优化器学习率	0.001
句子统一长度	34
Epoch	10
Batch_size	128
Dropout	0.5

Table 4. Confusion matrix
表 4. 混淆矩阵

预测值	实际值	
	Positive	Negative
Positive	TP	FP
Negative	FN	TN

其中，准确率代表正确计算的样本相对于分类样本的比例，计算公式如下：

$$A=\frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{TN}+\mathrm{FN}} \tag{29}$$

精确率是指预测结果为正例的统计中，被准确预测为正样本的比例，计算公式如下：

$$P=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{30}$$

召回率代表正确预测结果的正样本占全样本的实际正样本的比例，计算公式如下：

$$R=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{31}$$

F1 是精确率与召回率的加权平均数，统计公式如下：

$$\mathrm{F1}=2\times\frac{P\times R}{P+R} \tag{32}$$

4.4. 实验结果及分析

首先，本文进行两个实验，一采用 Mish 激活函数，另一个采用 Relu 激活函数，对比模型包括：BiLSTM、CNN、BiLSTM_CNN、BiLSTM_Attention、CNN_BiGRU_att [14]以及 SAttBiGRU_MCNN [15]。其对比结果如表 5 和表 6 所示：

结合表 5 和表 6，MAC_BiLSTM 文本分类模型较其他 6 种模型，在四个数据集都取得了较好的分类结果。通过表 5，本文给出的分类模型在四个数据集上准确率分别达到了 88.21%、87.41%、87.12%和 89.01%。对比 BiLSTM 模型，本文模型在四个数据集上准确率分别提升了 2.76%、1.61%、2.74%和 3.21%。比较 CNN 模型，本文模型在四个数据集上准确率分别增加了 4.33%、2.78%、4.07%和 4.39%。对比 BiLSTM_CNN 模型，本文模型在四个数据集上准确率分别提升了 2.02%、1.21%、0.80%和 2.99%。另外，

Table 5. Comparison of experimental results with Mish activation function
表 5. 加入 Mish 激活函数实验结果对比

文本分类模型	THUNews 数据集		今日头条数据集		os10c 数据集		Chn 数据集	
	准确率(%)	F1 值(%)	准确率(%)	F1 值(%)	准确率(%)	F1 值(%)	准确率(%)	F1 值(%)
BiLSTM	85.45	84.43	85.80	85.06	84.38	79.85	85.80	83.20
CNN	83.88	82.75	84.63	83.80	83.05	77.66	84.62	80.33
BiLSTM_CNN	86.19	85.32	86.20	85.54	86.32	82.29	86.02	82.86
BiLSTM_Attention	87.27	86.32	85.63	83.42	86.10	82.58	86.22	83.02
CNN_BiGRU_att	83.21	81.97	84.44	81.74	83.17	79.10	85.70	82.58
SAttBiGRU_MCNN	86.70	83.72	86.25	83.99	86.08	78.99	87.16	84.73
MAC_BiLSTM	**88.21**	**87.44**	**87.41**	**86.88**	**87.12**	**81.50**	**89.01**	**86.33**

Table 6. Comparison of experimental results with Relu activation function
表 6. 加入 Relu 激活函数实验结果对比

文本分类模型	THUNews 数据集		今日头条数据集		os10c 数据集		Chn 数据集	
	准确率(%)	F1 值(%)	准确率(%)	F1 值(%)	准确率(%)	F1 值(%)	准确率(%)	F1 值(%)
BiLSTM	85.29	84.36	84.61	82.23	83.74	73.07	85.30	82.67
CNN	80.71	79.56	83.73	82.89	81.21	75.78	84.15	80.81
BiLSTM_CNN	85.65	84.74	85.26	83.07	85.42	80.70	85.95	83.14
BiLSTM_Attention	86.31	85.43	84.94	84.29	84.88	80.38	85.68	81.92
CNN_BiGRU_att	81.94	80.93	82.49	79.85	82.30	78.27	84.72	82.09
SAttBiGRU_MCNN	85.39	84.25	85.35	84.59	85.95	81.98	86.87	84.51
MAC_BiLSTM	**87.09**	**86.14**	**86.16**	**85.51**	**86.78**	**80.69**	**87.40**	**84.93**

BiLSTM_CNN 模型比 BiLSTM 模型在四个数据集上的准确率分别提高了 0.74%、0.40%、1.94%和 0.22%，比 CNN 模型在四个数据集上准确率分别提高了 2.31%、1.57%、3.27%和 1.40%，表明 CNN 与 BiLSTM 结合的网络结构能更有效地提取文本中的关键特征，从而提高文本分类准确率。相比 BiLSTM_Attention 模型，本文模型在四个数据集上的准确率分别提高了 0.94%、1.78%、1.02%和 2.79%，并且，通过对比 BiLSTM 模型和 BiLSTM_Attention 模型的准确率可发现，自注意力机制的引入使模型的分类性能得到了进一步的改善。将本文所提模型与 CNN_BiGRU_att 模型和 SAttBiGRU_MCNN 模型的分类结果进行比较，发现本文模型比 CNN_BiGRU_att 模型准确率在四个数据集上分别增加了 5.00%、2.97%、3.95%和 3.31%，较 SAttBiGRU_MCNN 模型准确率在四个数据集上分别提高了 1.51%、1.16%、1.04%和 1.85%，表明本文提出的方法能够更充分地发挥出 CNN 与 BiLSTM 对文本特征的提取能力，并且本文模型在 CNN 通道中融合 BiLSTM 通道的输出，增强了特征的重用，进一步加强了 CNN 对文本局部特征信息的捕捉能力。同时，在 BiLSTM 通道和 CNN 通道中分别引入自注意力机制层和归一化层对通道输出的特征分布进行调整，增强了模型的学习能力，有效地提升了模型分类的准确性。

为探究各模型各评价指标与词向量维度之间的关系，本文进一步比较了各类模型在词向量维数依次为 50 维、100 维、150 维、200 维和 300 维下，MAC_BiLSTM 分类模型与其他深度学习分类模型词向量维度与准确率之间的关系，所得出的对比实验结果如图 5~8 所示：

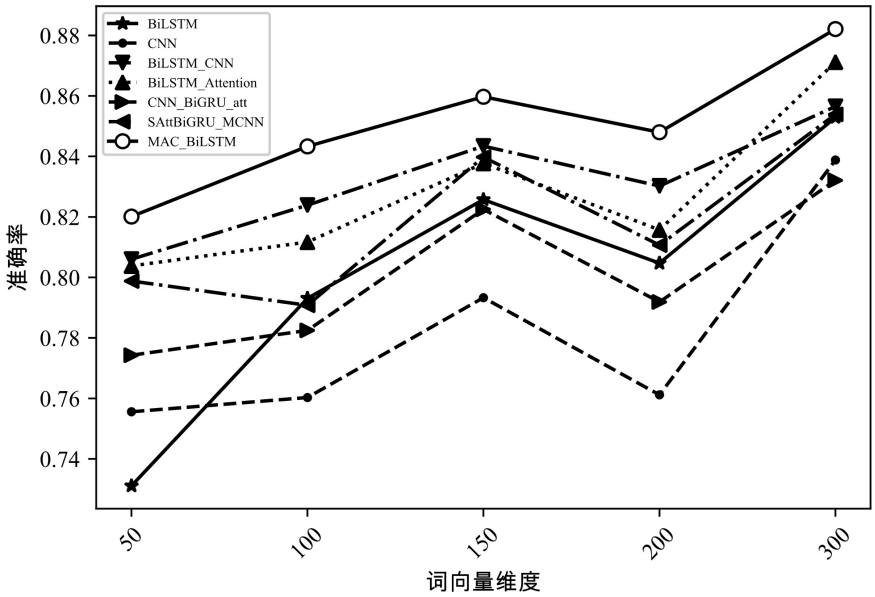

Figure 5. The accuracy comparison of the models under different word vector dimensions in THUC News dataset
图 5. THUC News 数据集不同词向量维度模型准确率对比

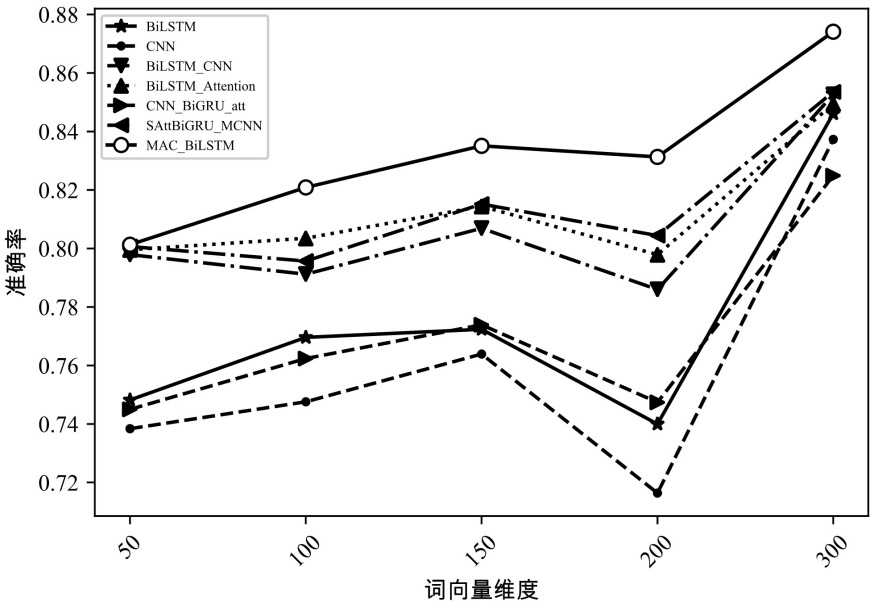

Figure 6. The accuracy comparison of the models under different word vector dimensions in Toutiao dataset
图 6. 今日头条数据集不同词向量维度模型准确率对比

根据图像，所有模型的准确率均伴随着词向量维度的上升，总体呈现出向上的趋势，并且各分类模型的分类性能在词向量维度为 300 维时实现最优。也就是随着词向量维度的增加，向量空间的表达能力增强，模型读取到的文本信息愈加全面、愈加丰富，从而各分类模型的性能均得到了提升。其中，本文模型在各个维度上的分类性能都优于其他主流深度学习模型，更证明了文中给出的 MAC_BiLSTM 模型在中文文本分类任务中的有效性与准确性。

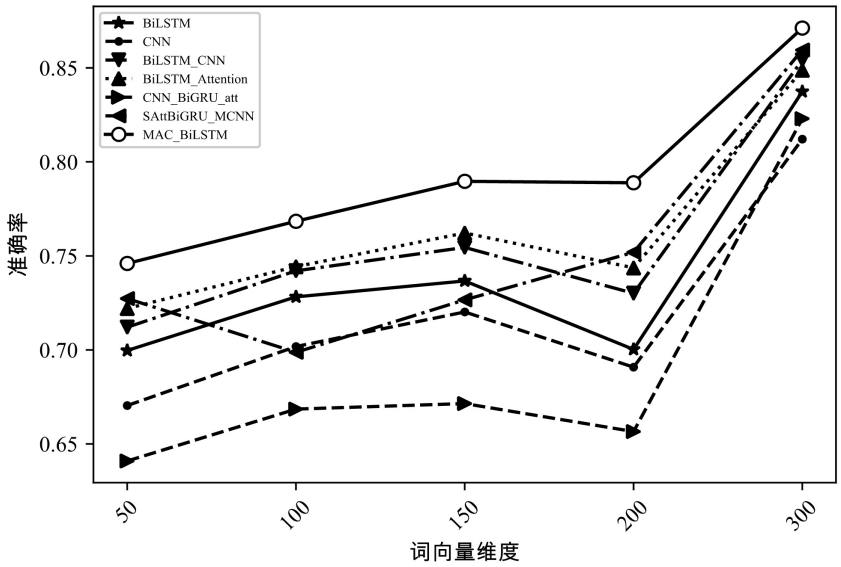

Figure 7. The accuracy comparison of the models under different word vector dimensions in os10c dataset
图 7. os10c 数据集不同词向量维度模型准确率对比

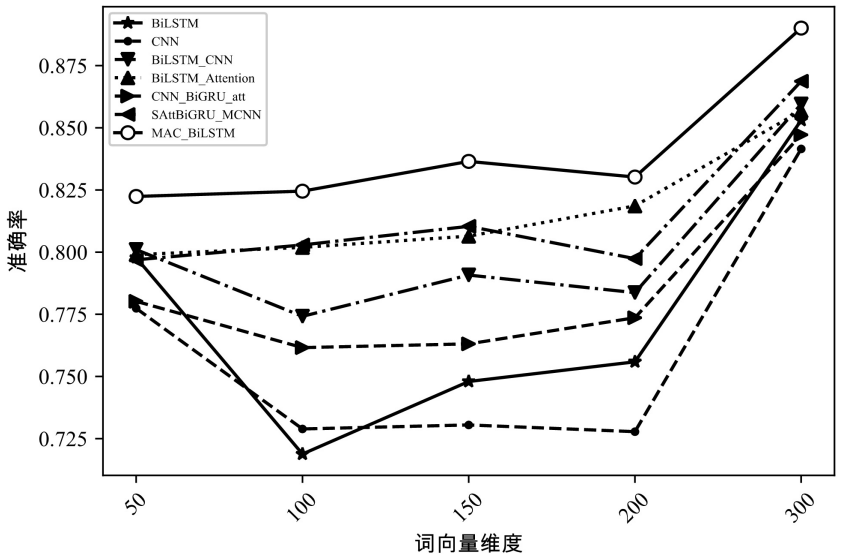

Figure 8. The accuracy comparison of the models under different word vector dimensions in Chn dataset
图 8. Chn 数据集不同词向量维度模型准确率对比

5. 结束语

本文结合 BiLSTM、CNN、自注意力机制以及归一化提出了多通道 MAC_BiLSTM 文本分类模型并对中文文本进行了分类研究。首先进行分词然后将文本数据增强，接着将文本数据通过 BiLSTM 对文本序列信息进行捕捉学习，提取文本不同层次的上下文语义信息。然后利用自注意力机制对文本深层次序列信息进行再提取，从而得到更加准确的文本关键语义信息，并将经过 BiLSTM 通道的信息与最初的词向量信息融合，输入 CNN 通道从而获得多特征的文本局部语义信息，得到更丰富的文本语义表示。通过将本文给出的文本分类模型，与其他六种模型在各个数据集上进行多维度的对比分析，结果显示本文提

供的模型有比较好的分类结果，从而证明本文模型的有效性，为自然语言处理领域提出了新的科研思路。

考虑到文本分类任务中词语的词向量表示对文本模型存在的影响，在接下来的研究中，将着力于在词汇的语义拓展和建模框架、参数等方面加以优化完善，以便于提高模型学习效果的准确率，并降低训练过程的时间成本。

致　谢

感谢全国统计科学研究项目对本论文的支持，感谢导师江开忠对本论文的指导，感谢作者杨洋、惠岚昕对本论文的协助与支持，感谢给予引用权文献与数据所有者对本论文的论据支撑。

基金项目

全国统计科学研究项目(2020LY080)。

参考文献

[1] Hochreiter, S. and Schmidhuber, J. (1997) Long Short-Term Memory. *Neural Computation*, **9**, 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

[2] Schuster, M. and Paliwal, K.K. (2002) Bidirectional Recurrentneural Networks. *IEEE Transactions on Signal Processing*, **45**, 2673-2681. https://doi.org/10.1109/78.650093

[3] Kim, Y. (2014) Convolutional Neural Networks for Sentence Classification. *Proceedings of the* 2014 *Conference on Empirical Methods in Natural Language Processing*, Doha, 25-29 October 2014, 1746-1751. https://doi.org/10.3115/v1/D14-1181

[4] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., et al. (2017) Attention Is All You Need. *Proceedings of the* 31*st Conference on Neural Information Processing Systems*, Long Beach, 4-9 December 2017, 5998-6008.

[5] Bengio, Y., Ducharme, R., Vincent, P. and Jauvin, C. (2003) A Neural Probabilistic Language Model. *The Journal of Machine Learning Research*, **3**, 1137-1155.

[6] Mikolov, T., Chen, K., Corrado, G. and Dean, J. (2013) Efficient Estimation of Word Representations in Vector Space. Computer Science. arXiv: 1301.3781.

[7] Zhang, J., Li, Y., Tian, J. and Li, T. (2018) LSTM-CNN Hybrid Model for Text Classification. 2018 *IEEE* 3*rd Advanced Information Technology, Electronic and Automation Control Conference* (*IAEAC*), Chongqing, 12-14 October 2018, 1675-1680. https://doi.org/10.1109/IAEAC.2018.8577620

[8] 吴汉瑜, 严江, 黄少滨, 李熔盛, 姜梦奇. 用于文本分类的 CNN_BiLSTM_Attention 混合模型[J]. 计算机科学, 2020, 47(z2): 23-27+34.

[9] 梁顺攀, 豆明明, 于洪涛, 郑智中. 基于混合神经网络的文本分类方法[J]. 计算机工程与设计, 2022, 43(2): 573-579.

[10] 张小川, 刘连喜, 戴旭尧, 刘璐. 基于词性特征的 CNN_BiGRU 文本分类模型[J]. 计算机应用与软件, 2021, 38(11): 155-161.

[11] 陶志勇, 李小兵, 刘影, 刘晓芳. 基于双向长短时记忆网络的改进注意力短文本分类方法[J]. 数据分析与知识发现, 2019, 3(12): 21-29.

[12] 蒲相忠, 梁春燕, 李鑫鑫, 赵磊, 王栋. 基于 Self-Attention 的多语言语义角色标注联合学习方法[J]. 计算机应用与软件, 2021, 38(12): 174-178.

[13] 邓朝阳, 仲国强, 王栋. 基于注意力门控图神经网络的文本分类[J]. 计算机科学, 2022, 49(6): 326-334.

[14] 陈农田, 李俊辉, 满永政. 基于改进 CNN-BiGRU-att 模型的文本分类研究[J/OL]. 昆明理工大学学报(自然科学版), 2022, 47(1): 30-37. https://doi.org/10.16112/j.cnki.53-1223/n.2022.01.131, 2021-09-28.

[15] 陈可嘉, 刘惠. 基于改进 BiGRU-CNN 的中文文本分类方法[J/OL]. 计算机工程, 2022, 48(5): 59-66+73. https://doi.org/10.19678/j.issn.1000-3428.0061176, 2021-12-11.

[16] Hinton, G.E., Ba, J.L. and Kiros, J.R. (2016) Layer Normalization. arXiv Preprint, arXiv: 1607.06450.

[17] Ioffe, S. and Szegedy, C. (2015) Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. arXiv Preprint arXiv: 1502.03167.

[18] Diganta, M. (2020) Mish: A Self Regularized Non-Monotonic Neural Activation Function. arXiv Preprint, arXiv: 1908.08681. https://arxiv.org/pdf/1908.08681.pdf

[19] THUCTC: 一个高效的中文文本分类工具包[OL]. http://thuctc.thunlp.org/, 2020-11-11.