

基于TextCRNN-OvR的患者咨询文本分类方法

张远芳

江南大学商学院, 江苏 无锡

收稿日期: 2023年3月24日; 录用日期: 2023年4月20日; 发布日期: 2023年4月27日

摘要

人工智能技术加速了互联网医疗发展, 患者在线问诊逐渐成为新趋势。然而大多数患者自身医学知识匮乏, 往往出现挂错科室的情况。因此, 患者咨询文本分类对于引导患者线上选择就诊科室显得十分重要。本文提出一种结合卷积循环神经网络与OvR策略的文本多分类方法, 既可以捕捉文本局部特征, 又可以学习词序信息。本文爬取了39问答网上的患者咨询文本作为数据源, 对所提方法进行了验证, 并与已有的分类算法作对比, 结果表明所提方法在精度、召回率、 F_1 值及准确率指标上具有更优越的算法性能。其中, 相较于其他SOTA的文本分类模型, TextCRNN-OvR在文本分类精度上取得了1%~4%不同程度上的提高, 这进一步说明了TextCRNN在提取文本特征方面以及本文OvR多分类策略的有效性。

关键词

在线问诊, 深度学习, TextCRNN模型, OvR策略, 文本多分类

Patient Consultation Text Classification Method Based on TextCRNN-OvR

Yuanfang Zhang

Business School, Jiangnan University, Wuxi Jiangsu

Received: Mar. 24th, 2023; accepted: Apr. 20th, 2023; published: Apr. 27th, 2023

Abstract

The development of Internet medical treatment has been accelerated by artificial intelligence technology, then online patient consultation is becoming a new trend. However, most patients often choose the wrong department due to a lack of adequate medical expertise. Therefore, the clas-

sification of patient consultation text is very important for guiding patients to choose departments online. This paper proposes a text multiple classification method combining the convolutional recurrent neural network and OvR strategy, which can capture local features of text but also learn word order information. In this paper, the proposed method is verified by crawling the patient consultation text on 39ask.com as the data source. Compared with existing classification algorithms, the results show that the proposed method has better performance in terms of precision, recall rate, F_1 score and accuracy. Among them, compared with other SOTA text classification models, TextCRNN-OvR has improved the accuracy of text classification by 1% to 4% to varying degrees, which further illustrates the advantages of TextCRNN in extracting text features and the effectiveness of the OvR multi-classification strategy in this paper.

Keywords

Online Inquiry, Deep Learning, TextCRNN Model, OvR Strategy, Text Multiclassification

Copyright © 2023 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

近年来, 各大医疗平台线上问诊人数持续增加。然而大部分患者在咨询时由于自身医疗专业知识的匮乏以及对自身症状的模糊认识, 往往不能选择正确的科室向医生提问[1], 因而如何高效智能化地将患者咨询文本进行准确的科室分类将对减轻医生负担、实现智能分诊具有重大意义。

近年来, 深度学习算法在计算机视觉、语音识别等领域得到广泛应用, 许多学者也开始逐渐将其转移到自然语言处理领域中, 并同样取得了较好的结果[2]。深度学习中常见的神经网络模型包括卷积神经网络(Convolutional Neural Network, CNN)、循环神经网络(Recurrent Neural Network, RNN)等。CNN 不能很好处理文本时序性问题, 而 RNN 刚好考虑到与时间序列有关的问题, 擅长于捕捉文本序列中的长期依赖关系[3]。因而 Zhou 等人[4]为整合二者优势, 提出了卷积循环神经网络模型(Convolutional Recurrent Neural Network, CRNN)并用于文本分类, 结果证明 TextCRNN 模型在分类准确率及其他评价指标上高于已有的 CNN 模型和 RNN 模型。

本文提出了将 TextCRNN 模型与 OvR 策略结合的文本分类方法, 并爬取 39 问答网上的患者咨询文本作为实验数据与其他常见深度学习模型进行对比, 结果表明本文所提方法取得了更为精准的分类结果, 对促进智能分诊、实现精准导医具有理论指导意义。

2. 相关研究

随着深度学习技术的发展, 文本分类的研究趋势逐渐由基于传统机器学习模型的算法转向基于深度学习模型的算法上, 一般主要有基于 TextCNN 模型和基于 TextRNN 模型的文本分类方法。

在基于 TextCNN 模型的文本分类任务上, Kim [5]提出了一个单层卷积层结构的 CNN 模型, 通过采用多个卷积核提取文本特征, 最后通过池化拼接得到特征向量, 在一系列分类任务上取得较好效果。Kalchbrenner 等人[6]提出了一个双层卷积层结构的 DCNN 模型, 通过动态池化策略和多层卷积使得文本句子中相隔较远的词也有语义联系, 从而在许多情感分类、问题分类等任务上取得满意结果。

在基于 TextRNN 模型的文本分类任务上, 许多学者使用长短期记忆网络(Long Short-Term Memory,

LSTM)和门控循环单元(Gated Recurrent Unit, GRU)居多,两者均是特殊的 RNN 模型。有学者尝试将 CNN 模型与 RNN 模型结合,而 Xiao 等人[7]提出了结合顺序相反的 TextCRNN 模型,即先通过 CNN 模型的多层卷积网络层,再将生成的特征输入到 RNN 模型的双向 LSTM 层进行处理。

回顾以往的相关研究,可以看出 TextCNN 与 TextRNN 模型在文本分类问题上得到了广泛应用与发展,有效避免了复杂的文本特征工程,简化了传统文本分类方法的繁杂文本预处理工作,且具有比传统方法更优越的算法性能。因此本文考虑结合 TextCRNN 模型及 OvR 策略运用到患者咨询文本分类任务上,可以同时兼顾两类深度学习模型的优势,从而降低模型预处理成本且可以进一步提升模型分类精确程度。

2. 基于 TextCRNN-OvR 的文本分类模型

2.1. TextCRNN 模型

TextCRNN 模型结合了 CNN 在捕获空间和时间结构信息的局部相关性的优点,以及 RNN 在捕获任意长度的序列结构信息的长期依赖关系的优点,从而在许多文本分类任务中均获得较高的分类效果[8]。TextCRNN 模型主要由包括卷积层、池化层、RNN 层(BiLSTM 层或 BiGRU 层)、全连接层和 Softmax 层。

2.1.1. 卷积层

卷积层主要先对初始数据作卷积运算从而降低模型复杂度。假设文本数据中经过预处理的特征词向量为 $a_i \in T^d$, 表示第 i 个特征词的 d 维向量,则长度为 n 的句子可表示为

$$A_{1:n} = a_1 \oplus a_2 \oplus \cdots \oplus a_n \quad (1)$$

其中, $\oplus$ 为拼接操作,若干个词向量 $a_k, a_{k+1}, \dots, a_{k+l}$ 拼接而成的句子可表示为 $A_{k:k+l}$ 。假设卷积核为 $\omega \in T^{h \times d}$, 其中 h 为卷积核的宽度,表示每次捕捉 h 个词进行计算。当输入含 h 个词的句子 $A_{k:k+h-1}$ 时,经过卷积运算后产生一个新的特征,可表示为

$$c_k = f(\omega \otimes A_{k:k+h-1} + b) \quad (2)$$

其中, $\otimes$ 为卷积操作, b 表示偏置量, $f(x)$ 为非线性激活函数,常用的激活函数为修正线性单元(Rectified Linear Unit, ReLU),可表示为

$$f(x_i) = \begin{cases} x_i, & \text{if } x_{ii} > 0 \\ 0, & \text{if } x_i \leq 0 \end{cases} \quad (3)$$

在实际模型训练中,句子 $A_{k:k+h-1}$ 不同个数相邻词组成的短句之间特征不同,将宽度为 h 的卷积核去捕捉该句子所有可能组成的短句,从而可获得特征向量为

$$C = [c_1, c_2, \dots, c_k] \quad (4)$$

2.1.2. 池化层

在实际模型训练中,通常采用若干不同卷积核进行卷积操作,随后进行池化运算,从而可得到每个特征向量中的最重要特征值。

假设采取了 J 个卷积核进行卷积运算,对结果进行最大池化即 $\bar{C} = \max \{C\}$,再将池化后的结果进行拼接运算,即可得到

$$C_M = [\bar{C}_1, \bar{C}_2, \dots, \bar{C}_J] \quad (5)$$

2.1.3. RNN 层

RNN 层通常由 LSTM 层或者 GRU 层组成。LSTM 模型是一个记忆存储单元,主要包括三组门单元:

输入门、遗忘门和输出门。通过自身门控状态控制传输状态，有选择性的记忆重要信息，从而很好的解决文本分类任务中常见的长期依赖问题[9]。GRU 是 LSTM 的变种模型，通过组合遗忘门和输入门到一个单独的门，使得模型比 LSTM 更简单，参数更少从而使得模型训练难度减少了很多[10]。

然而单向的 LSTM 和 GRU 模型不能很好地利用文本数据中上下文信息，从而有学者提出相应的 BiLSTM 模型[11]与 BiGRU 模型[12]。BiLSTM 模型能够提取文本的上下文关联关系和指代关系，从而提高模型训练效果。模型中各个门单元和状态之间的公式为

$$input_t = f(W_i \otimes [p_{t-1}, q_t] + b_i) \quad (6)$$

$$forget_t = f(W_f \otimes [p_{t-1}, q_t] + b_f) \quad (7)$$

$$output_t = f(W_o \otimes [p_{t-1}, q_t] + b_o) \quad (8)$$

$$p_t = output_t \times \tanh(S_t) \quad (9)$$

$$\overline{S}_t = \tanh(W_s \otimes [p_{t-1}, q_t] + b_s) \quad (10)$$

$$S_t = forget_t \times S_{t-1} + input_t \times \overline{S}_t \quad (11)$$

其中， $input_t$ 为输入门函数， $forget_t$ 为遗忘门函数， $output_t$ 为输出门函数， W_i ， W_f ， W_o ， W_s 分别表示三个门函数及状态更新的权重， b_i ， b_f ， b_o ， b_s 则分别为其对应的偏置量。 p_t 表示 t 时刻 LSTM 层隐藏状态的更新信息， q_t 为前面池化层输出的特征向量。 $\overline{S}_t$ 表示 t 时刻输入的候选信息， S_t 表示 t 时刻记忆单元状态。池化层输出结果经过双向 BiLSTM 层后得到具有前、后向特征的文本特征向量，随后通过全连接层进行再次拼接得到新的特征向量。其中通过的全连接层进行的运算实际上为作卷积核大小为上层特征大小的卷积运算。为提升模型泛化效果，我们在全连接层之后再加入 Dropout 层，该层以一定概率值随机减少中间特征的数量[13]，从而防止模型过拟合。

2.1.4. Softmax 层

在经过 Dropout 层后，将得到的特征向量输入最后的 Softmax 层进行类别预测。假设输入特征向量为 C_M_B ，文本样例数为 m ，文本类别数为 n ，Softmax 层将输出各类别的预测概率值

$$P = \text{Softmax}(\omega \otimes (C_M_B \odot r) + b) \quad (12)$$

其中 $\odot$ 表示按元素的乘法运算， r 是掩盖向量。Softmax 函数使得该层所有输出概率值和为 1。

在文本分类模型训练时，通常使用类别交叉熵(Categorical Cross Entropy)作为损失函数，为了防止结果过拟合，我们通常加入正则项，因而该函数定义为

$$loss_{category} = \sum_m \sum_n T_m^n \log Y_m^n + \lambda \|\phi\|^2 \quad (13)$$

其中， T_m^n 表示样例的真实类别， Y_m^n 表示样例的预测类别， λ 为正则项的系数， ϕ 为模型的参数集合。

2.2. 基于 OvR 策略的分类模型

本文选取多分类任务中常使用的 OvR 策略，该策略主要思路为在训练时每次将某个类别的样例作为正例，所有其他类别的作为反例，在测试时如果仅有一个分类器判断为正例，则对应分类器类别标记为最终结果，如果多个分类器预测为正例，则考虑各个分类器的预测概率值，选择概率值较大的类别标记为最终结果。

2.3. 基于 TextCRNN-OvR 的文本分类模型

本文构建了基于 OvR 策略的 TextCRNN 文本分类模型，主要分为文本预处理阶段、TextCRNN 分类

器构建阶段与分类器预测结果确定阶段，模型框架如图 1 所示。

首先，将文本数据进行简单的预处理，通过删去重复值、去标点等无关符号、分词、去停用词得到预处理文本，再通过随机初始化得到文本的词向量矩阵。即可作为后面 TextCRNN 模型的输入；其次，在单个 TextCRNN 模型分类器训练阶段，先令文本词向量矩阵通过 CNN 层，再通过 RNN 层。由于双向 LSTM 模型在文本分类任务上效果普遍优于双向 GRU 模型[14]，因此本文采取了双向 LSTM 模型作为 RNN 层。重复多个模型训练过程即可得到多个不同类别的二分类器；最后，在分类器预测阶段，将测试数据集依次导入基于 OvR 策略的各个分类器中，根据每个分类器的类别预测概率值进行最终类别判定。

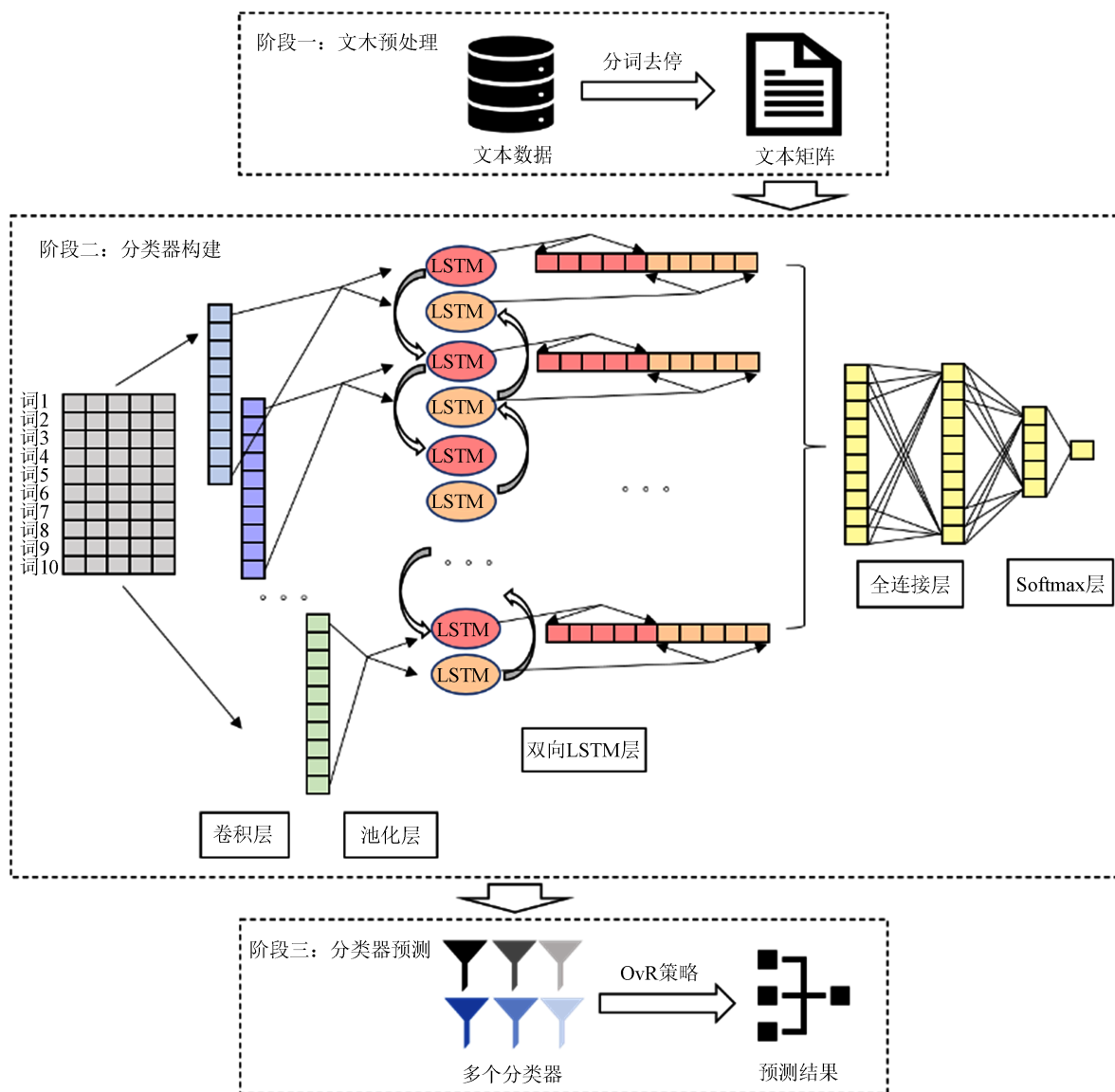


Figure 1. Text classification model framework based on TextCRNN-OvR

图 1. 基于 TextCRNN-OvR 的文本分类模型框架

本文所提模型实现过程可根据上述框架也分为相应的三个阶段。模型输入为带类别标签的文本，输出为用于文本分类可视化的混淆矩阵，为后续模型评价作参考。模型核心算法伪代码如下：

输入：带类别标签文本 $D = \{(x_{11}, y_1), \dots, (x_{1M}, y_1); (x_{21}, y_2), \dots, (x_{2N}, y_2); \dots, (x_{k1}, y_k), \dots, (x_{kO}, y_k)\}$ (设 d 为文本总数, k 为类别数);

输出：文本分类混淆矩阵 $M_{k \times k}$;

1: step=1;

2: for $i = 1, 2, \dots, d$ **do**

3: 去标点等特殊符号、分词、去停用词、生成词向量;

4: endfor

5: 输出文本预处理后的词向量集 $\bar{D} = \{a_{11}, \dots, a_{1m}; a_{21}, \dots, a_{2n}; \dots, a_{k1}, \dots, a_{ko}\}$;

6: step=2;

7: 初始化 TextCRNN 参数, 读入数据集 $\bar{D}$ 并随机将其打乱, 取 80% 作训练集;

8: for $i = 1, 2, \dots, k$ **do**

9: 根据式(2)进行卷积运算, 并对结果进行最大池化运算;

10: 根据式(6)-(11)进行 RNN 层运算, 将结果拼接后输入 Dropout 层;

11: 训练模型并保存;

12: endfor

13: step=3;

14: 读入数据集 $\bar{D}$ 并随机将其打乱, 取 20% 作测试集;

15: for $i = 1, 2, \dots, k$ **do**

16: 将测试数据集依次输入 k 个二分类模型中;

17: 根据式(12)计算 k 个类别的预测概率 $\{P_{10}, P_{11}, P_{20}, P_{21}, \dots, P_{k0}, P_{k1}\}$;

18: if $P_{i1} > P_{i0} \ \& \ P_{(i+1)1} \leq P_{(i+1)0} \ \& \dots \ \& \ P_{k1} \leq P_{k0}$

19: 将 i 对应的类别作为预测类别;

20: else 计算 $j^* = \arg \max_j \{P_{j1}, \dots, P_{k1}\}$, 将 j^* 对应的类别作为预测类别;

21: endfor

3. 实例研究

3.1. 实验数据

本文的实验数据来源于“39 问医生”网站(<http://ask.39.net/>), “39 问医生”平台有超过 14 万名医生注册, 收录了超过 28,017 万条患者咨询信息。图 2 为血液科的患者历史提问界面。

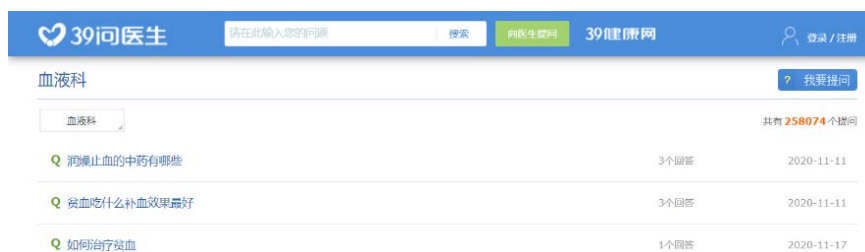


Figure 2. Patient history questioning interface on the “39 Ask Doctors” website

图 2. “39 问医生”网站患者历史提问界面

我们基于 Python 语言编写爬虫代码爬取该网站上 6 个常见科室(内科、外科、儿科、男科、妇产科、传染病科)的患者咨询问题数据。“39 问医生”网站上每页共 32 个患者提问, 6 个科室共爬取 8400 页, 共计 268,800 条提问。首先我们对数据进行简单的筛选, 通过删除文本字数在 5 以下以及重复提问等的患者提问, 最终剩余 248,991 条提问用于后续研究。各科室的提问示例与数量如表 1 所示。

Table 1. Experimental data set
表 1. 实验数据集

科室	提问示例	提问条数
内科	吃东西下咽到胸口正中间的时候感觉疼痛, 是怎么回事?	38,904
外科	腰闪了, 行动困难, 用红外线理疗灯可以帮助恢复吗?	44,397
儿科	婴儿七个月, 吐奶带有好多血丝, 是什么原因?	46,685
男科	前列腺钙化和增生有什么区别, 饮食需要注意什么?	41,836
妇产科	慢性盆腔炎肚子疼痛难忍, 怎样才能缓解疼痛?	35,086
传染病科	之前得过单纯疱疹, 现在身上又起红点怎么回事?	42,083
总计		248,991

其中, 我们通过对患者提问字数进行统计分析, 如图 3 所示。在 248,991 条提问中, 大部分提问字数基本在 20 左右。本文将各个科室的数据集随机打乱, 选取各科室 80%提问文本作为训练集, 20%作为测试集。为减少实验结果的随机性, 本文重复数据集选取过程 20 次, 每次选取过程确保随机性, 最终取实验结果的平均值来评价所提新模型的分类效果以及与其他已有模型的对比。

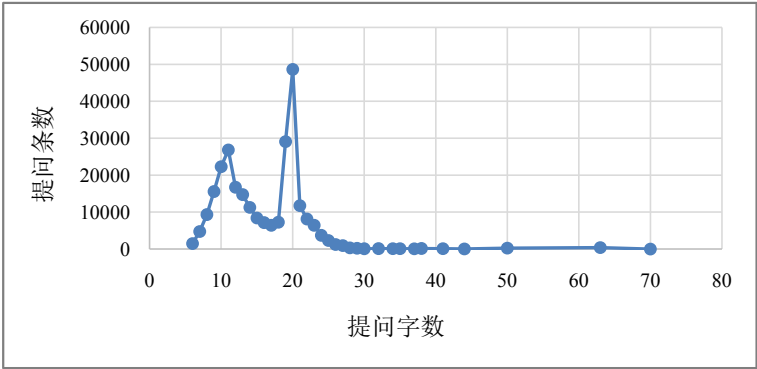


Figure 3. Statistical chart of the frequency of patient questions
图 3. 患者提问字数频率统计图

3.2. 评价指标

对于分类问题, 可以根据测试集样本上的真实类别与分类器预测的类别的组合分为真正例(True Positive, TP)、真反例(True Negative, TN)、假正例(False Positive, FP)和假反例(False Negative, FN)。我们用 Num_TP 、 Num_TN 、 Num_FP 和 Num_FN 分别表示以上 4 种组合的样例数, 从而可以定义分类问题中评价分类效果常用的评价指标: 精度(Precision)、召回率(Recall)和 F_1 值(F_1 -Score)。三者的计算公式如下:

$$Precision = Num_TP / (Num_TP + Num_FP) \quad (14)$$

$$Recall = Num_TP / (Num_TP + Num_FN) \quad (15)$$

$$F_1\text{-Score} = 2 \times Precision \times Recall / (Precision + Recall) \quad (16)$$

从上述三个公式不难看出, 精度表示真实类别为正例的样例占有所有被预测为正例样例的比率, 召回率表示真实正例占有所有真实类别为正例样例中预测准确的比率, 二者相互联系又相互约束。\$F_1\$ 值则是综合考虑了精度和召回率的评价指标。除了上面三个常用的评价指标外, 还有一个最直观的指标准确率 (Accuracy), 它表示分类中预测正确的样例数占总样例的比率, 计算公式如下:

$$Accuracy = (Num_TP + Num_TN) / (Num_TP + Num_TN + Num_FP + Num_FN) \quad (17)$$

分析式(14)~式(17), 可知 4 个评价指标的取值范围都在[0,1], 我们希望上述四个指标值越大越好, 即表示分类效果越好。

3.3. 参数设置

本文基于 TensorFlow 实现框架构建 TextCRNN 模型, 训练文本最大长度设置为 70, 模型 Embedding 层采用的词向量维度取 300。模型卷积层批处理尺寸取 256, 卷积核尺寸设置为 2, 3, 4, 5, 各个卷积层的卷积核数量设置为 300。Dropout 层参数(随机失活概率值)设置为 0.5。损失函数采用对数损失函数 (Categorical Crossentropy), 优化算法采用 Adam 算法, 其中, Adam 算法的步长 \$\epsilon\$ 取 \$10^{-3}\$, 矩估计的指数衰减速率 \$\rho_1\$ 和 \$\rho_2\$ 分别取 0.9 和 0.999, 用于数值稳定的小常数 \$\delta\$ 取 \$10^{-8}\$ [15]。模型训练步长为 5000, 训练轮次为 20, 最后设置准确率作为保存更好模型的评价标准。

实验运行环境为: CPU 型号为 i5-10210U, 主频 2.1 GHz, 内存为 8 GB, GPU 型号为 MX250, 操作系统为 Windows10, Python 版本为 3.5, 运行软件为 PyCharm。

3.4. 实验结果

为更好验证模型的有效性, 本文选取了其他常见的深度学习方法(FastText、TextCNN、TextRNN (BiLSTM)、TextRNN (BiGRU)、TextRCNN)作对比实验。

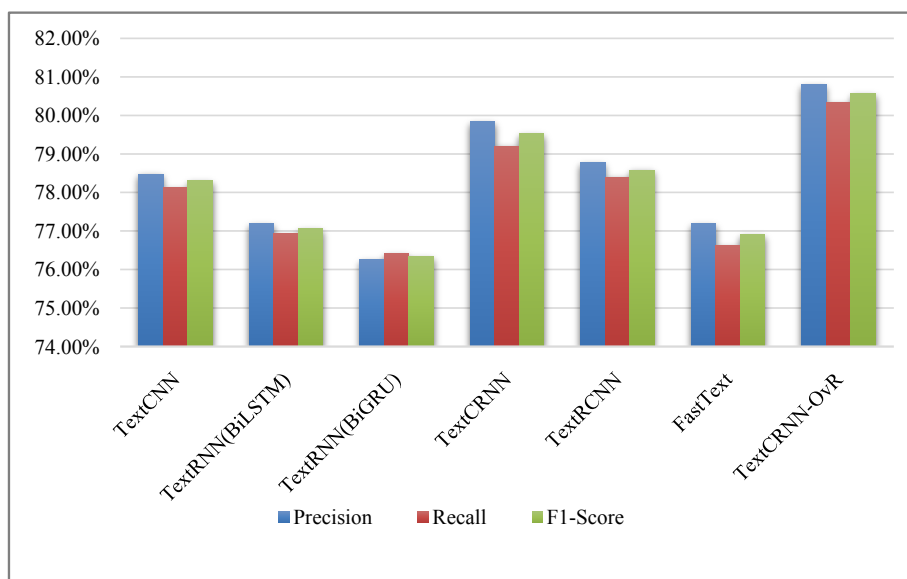


Figure 4. Text classification model evaluation indicator chart

图 4. 文本分类模型评价指标图

本文将新提的 TextCRNN-OvR 模型与其他深度学习模型进行对比,模型在精度、召回率、 F_1 值上的实验结果如图 4 所示。从图中可以看出,所提新模型在三个评价指标上均达到 80%以上,相比已有其他模型都有明显提高。各个模型在准确率上的实验结果如表 2 所示,从表中可以看出 TextCRNN-OvR 模型的文本分类准确率达到 80.63%,相比其他模型也得到了较大提高。

Table 2. Experimental results of accuracy of text classification model
表 2. 文本分类模型准确率实验结果

模型	准确率
TextCRNN-OvR	80.63%
FastText	76.87%
TextRCNN	78.59%
TextCRNN	79.55%
TextRNN (BiLSTM)	77.08%
TextRNN (BiGRU)	76.45%
TextCNN	77.49%

4. 结束语

本文将深度学习模型应用到患者咨询文本分类问题中,从而可以根据患者的咨询问题为其推荐准确的科室。本文主要研究结果有: 1) 构建了一个结合 TextCRNN 模型与 OvR 策略的分类模型,该模型不仅具有 CNN 模型在挖掘文本局部特征上的优势,以及 RNN 模型在获取文本前后时序关系上的优势,且可以通过 OvR 策略进一步提高分类效果; 2) 本文所提的 TextCRNN-OvR 模型在患者咨询文本分类任务中相比于其他深度学习模型在精度、召回率、 F_1 值和准确率上均得到了明显提高,首先,相较于其他 SOTA 的文本分类模型,TextCRNN 的分类准确率是最高的,这说明 TextCRNN 在提取文本特征方面是很有效的,同时相比于其他模型,TextCRNN-OvR 在分类精度上取得了 1%~4%不同程度上的提高,这说明了本文 OvR 多分类策略的有效性; 3) 本文所提患者咨询文本分类模型可为互联网医疗智能分诊、精准导医提供决策支持,对于落实分级诊疗也具有重要的意义。

在本文的研究中,我们对爬取到的文本没有进行深入预处理,只是简单的分词去停,以后的研究工作中可考虑引入基于 Word2Vec 或 Bert 的词嵌入模型。另外,由于咨询文本普遍较短,如何从海量短文本中提取特征也是未来值得研究的工作重点。

参考文献

- [1] 王若佳, 张璐, 王继民. 基于机器学习的在线问诊平台智能分诊研究[J]. 数据分析与知识发现, 2019, 3(9): 88-97.
- [2] 何炎祥, 孙松涛, 牛菲菲, 李飞. 用于微博情感分析的一种情感语义增强的深度学习模型[J]. 计算机学报, 2017, 40(4): 773-790.
- [3] Abdi, A., Shamsuddin, S.M., Hasan, S., et al. (2019) Deep Learning-Based Sentiment Classification of Evaluative Text Based on Multi-Feature Fusion. *Information Processing and Management*, **56**, 1245-1259.
<https://doi.org/10.1016/j.ipm.2019.02.018>
- [4] Zhou, C., Sun, C., Liu, Z., et al. (2015) A C-LSTM Neural Network for Text Classification. *Computer Science*, **1**, 39-44.
- [5] Kim, Y. (2014) Convolutional Neural Networks for Sentence Classification. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Stroudsburg, 26-28 October 2014, 1746-1751.
<https://doi.org/10.3115/v1/D14-1181>
- [6] Kalchbrenner, N., Grefenstette, E. and Blunsom, P. (2014) A Convolutional Network for Modeling Sentences. *Pro-*

- ceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, Volume 1, 655-665.
<https://doi.org/10.3115/v1/P14-1062>
- [7] Xiao, Y. and Cho, K. (2016) Efficient Character-Level Document Classification by Combining Convolution and Recurrent Layers.
 - [8] Huang, X., Qiao, L., Yu, W., *et al.* (2020) End-to-End Sequence Labeling via Convolutional Recurrent Neural Network with a Connectionist Temporal Classification Layer. *International Journal of Computational Intelligence Systems*, **13**, 341-351. <https://doi.org/10.2991/ijcis.d.200316.001>
 - [9] 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>
 - [10] Cho, K., Van Merriënboer, B., Gulcehre, C., *et al.* (2014) Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, October 2014, 1724-1734. <https://doi.org/10.3115/v1/D14-1179>
 - [11] 王伟, 孙玉霞, 齐庆杰, 孟祥福. 基于 BiGRU-attention 神经网络的文本情感分类模型[J]. 计算机应用研究, 2019, 36(12): 3558-3564.
 - [12] Han, Y., Liu, M. and Jing, W. (2020) Aspect-Level Drug Reviews Sentiment Analysis Based on Double BiGRU and Knowledge Transfer. *IEEE Access*, **8**, 21314-21325. <https://doi.org/10.1109/ACCESS.2020.2969473>
 - [13] Shilaskar, S., Ghatol, A. and Chatur, P. (2017) Medical Decision Support System for Extremely Imbalanced Datasets. *Information Sciences*, **384**, 205-219. <https://doi.org/10.1016/j.ins.2016.08.077>
 - [14] Zhou, P., Qi, Z., Zheng, S., *et al.* (2016) Text Classification Improved by Integrating Bidirectional LSTM with Two Dimensional Max Pooling.
 - [15] Kingma, D.P. and Ba, J.L. (2015) Adam: A Method for Stochastic Optimization. *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015)*, San Diego, 7-9 May 2015. arXiv:1412.6980