基于堆叠模型的司法短文本多标签分类

何 涛[1],陈 剑[1],闻英友[1],孔为民[2]
(1.东北大学 东软研究院,辽宁 沈阳 110169;
2.定陶区人民检察院,山东 菏泽 274100)

摘 要:司法文书短文本的语义多样性和特征稀疏性等特点,对短文本多标签分类精度提出了很大的挑战,传统单一模型的分类算法已无法满足业务需求。为此,提出一种融合深度学习与堆叠模型的多标签分类方法。该方法将分类器划分成两个层次,第一层使用BERT、卷积神经网络、门限循环单元等深度学习方法作为基础分类器,每个基础分类器模型通过K折交叉验证得到所有数据的多标签分类概率值,将此概率值数据进行融合形成元数据;第二层使用自定义的深度神经网络作为混合器,以第一层的元数据为输入,通过训练多标签概率矩阵获取模型参数。该方法将强分类器关联在一起,获得比单个分类器更加强大的性能。实验结果表明,深度学习堆叠模型实现了87%左右的短文本分类F1分数,优于BERT、卷积神经网络、循环神经网络及其他单个模型的性能。

关键词:堆叠模型;BERT;卷积神经网络;门限循环单元;多标签分类
中图分类号:TP 391 **文献标识码**:A **文章编号**:1673-629X(2021)03-0027-06
doi:10.3969/j.issn.1673-629X.2021.03.005

Multi-label Classification of Judicial Short Texts Based on Stacking Model

HE Tao[1],CHEN Jian[1],WEN Ying-you[1],KONG Wei-min[2]
(1.Neusoft Research,Northeastern University,Shenyang 110169,China;
2.People's Procuratorate of Dingtao,Heze 274100,China)

Abstract:The semantic diversity and feature sparsity of short texts in judicial documents is a great challenge to the accuracy of multi-label classification,so the traditional single model classification algorithm can no longer meet the business needs. For this reason,we propose a multi-label classification method combining deep learning and stacking model. This method divides the classifiers into two layers. In the first layer,deep learning methods such as BERT,convolutional neural network and gated recurrent unit are used as the basic classifier. Each basic classifier model obtains the multi-label classification probability value of all data through K-fold cross-validation,which are merged to form metadata. In the second layer,the user-defined deep neural network is used as the mixer,and the metadata in the first layer is used as the input,and the model parameters are obtained by training the multi label probability matrix. This method associates the strong learners together and gains more powerful functions than a single classifier. The experiment shows that the proposed model stacking method achieves about 87% of the F1 score of short text classification,which is superior to BERT,convolutional neural network,cyclic neural network and other single models.

Key words:stacking model;bidirectional encoder representations from transformers;convolutional neural network;gated recurrent unit;multi-label classification

0 引 言

随着国内司法业务信息化的发展,司法领域产生了巨量的文本数据,目前办案人员主要依靠手工分析案件卷宗、提取案件要素的工作方式,效率低下,已无法满足智慧司法的客观需要。如何在海量司法文书数据中自动抽取出有价值的信息,具有巨大的社会意义和商业价值。一种可行的方式是将司法文书进行细粒度分割,生成短文本子集,并通过深度学习等智能化方法对短文本进行多标签分类,将案件要素抽取出来呈现给办案人员。高效地将大规模司法短文本数据进行

收稿日期:2020-04-16 **修回日期**:2020-08-20
基金项目:国家重点研发计划(2018YFC0830601);辽宁省重点研发计划(2019JH2/10100027);教育部基本科研业务费项目(N171802001);辽宁省"兴辽英才计划"项目(XLYC1802100)
作者简介:何 涛(1981-),男,硕士,高级工程师,研究方向为自然语言处理、计算机视觉。

正确的归类,是智慧司法系统的基本任务,也是其他司法过程的基础。

近年来,随着计算能力和深度学习算法的快速发展,深度学习在人工智能的多个领域都取得了显著的进展。通过使用非线性网络结构实现复杂的函数表达,并在特征表达时使用分布式特征输入,使深度学习凭借强大的特征学习能力,在自然语言处理领域取得令人瞩目的成绩。

Arevian[1]使用真实世界中的文本对循环神经网络进行训练,完成文本分类。Chen 等人[2]采用扩展短文本特征的方式派生特定粒度的暗含主题,并在在多个主题粒度上,利用多主题来更精确的进行短文本建模。Fu 等人[3]使用卷积神经网络对司法文书进行分类,达到了比传统的基于 Logistic 回归和支持向量机更好的效果。Kim[4]提出的 TextCNN 模型在文本分类方面取得了很好的效果,使得该模型成为 CNN 在自然语言处理中应用最广泛的模型。Kalchbrenner 等[5]提出了动态的 k-max pooling 机制,使得文本特征提取能力进一步增强。Lei 等[6]在标准卷积层使用基于张量的词间操作代替串接词向量的线性运算。Zhang 等人[7]使用 N-Gram 模型扩展短文本,通过词语之间的相似度阈值判定文本的分类。陈钊等人[8]使用情感词典识别构成二值特征作为外部辅助特征,提高了 CNN 模型的处理能力。Shi 等人[9]提出了卷积循环神经网络,在处理序列对象时比传统神经网络模型具有一些优点。Vaswani 等人[10]提出基于多头自注意力机制的 Transformer 模型,大大提高了文本特征提取能力,为序列标注任务提出新的解决方法。Yang 等[11]在 LSTM 模型运用 Attention 机制进行文本级分类,取得了较好的分类效果。Xiao 等人[12]提出结合卷积神经网络和循环神经网络的方式提取文本特征,结合了两种神经网络的特点。Hassan 等人[13]针对卷积神经网络在捕获文本特征长期依赖问题时需要多层网络,提出联合 CNN 和 RNN 网络模型。Yin 等人[14]提出一个更为细化的卷积神经网络 ATTCONV,该算法使用注意力机制扩展了卷积运算的上下文范围。2018 年 10 月底,Google 公布 BERT(bidirectional encoder representation from transformers)[15]预训练模型在 11 项 NLP 任务中刷新纪录,引起业界的广泛关注。

然而,现有的方法应用于司法短文本多标签分类时,还存在分类准确率不高的问题,主要原因是提取文本特征的方式仍然过于单一。为此,该文提出了一种基于深度学习堆叠模型的多标签分类方法,融合了 Transformer、卷积神经网络、循环神经网络等各种深度学习算法的优势,解决了提取特征角度单一的问题,进一步提升了短文本多标签分类性能。

1 短文本分类堆叠模型

短文本多标签分类问题可以定义为:对于给定的样本数据集 $D=\{X,Y\}$,其中 $X=\{x_i\}_{i=1}^{N}$ 表示该数据集的样本空间,N 为样本空间中样本的数量,$Y=\{y_i^1,y_i^2,\cdots,y_i^Q\}_{i=1}^{N}$ 表示样本集的标签空间,Q 为标签空间中标签的数量。x_i 是样本集中的第 i 个实例,即一个短文本,将其进一步细化为 $x_i=\{w_1,w_2,\cdots,w_m\}$,$w_i$ 表示一个实例中第 i 个字符的词向量。y_i 的取值范围为 $\{0,1\}$,0 表示样本不属于 y_i 类别,1 表示样本属于 y_i 类别。

提出的堆叠模型整体架构如图 1 所示。

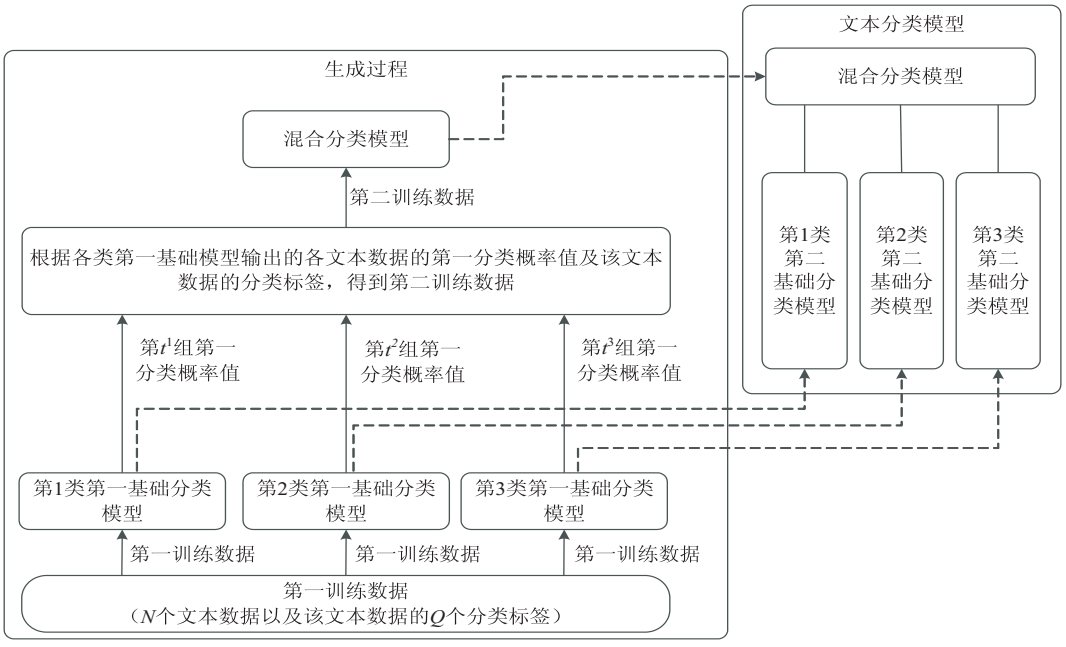

图 1 短文本分类堆叠模型整体架构

首先使用 K 折交叉验证分割训练数据集,在第一层的分类模型上分别得到预测模型: $F_j(X^i) \rightarrow \boldsymbol{P}_j^i$, $\boldsymbol{P}_j^i$ 是交叉验证过程中第 j 个分类模型的第 i 折作为验证集生成的多标签概率矩阵。将 $\boldsymbol{P}_j^i$ 进行拼接,作为混合分类器的输入数据,训练混合分类器,得到: $G(P_{1,2,\cdots,q}^i) \rightarrow L^i$,其中 L^i 为第 i 个样本的多标签分类结果。该文以第一层使用 3 个分类模型为例。

这种模型的优势在于,首先分别使用 3 种不同类型的深度学习网络从不同角度提取文本特征,其次将不同模型的输出结果以多标签概率值进行融合,可以获得比分类结果更加丰富的信息。第一基础分类模型与第二基础分类模型使用相同的结构、相同的训练参数,只是使用了不同的训练数据,应用在不同的过程中。

为验证该方法的有效性,第一层分类模型分别使用 BERT 预训练模型、单通道 TextCNN 模型、Bi-GRU 模型,混合分类模型使用自定义的包含两个隐藏层的深度神经网络。

1.1　BERT 预训练模型

BERT 预训练模型是在多层 Transformer 编码器的基础上实现的。Transformer 编码器作为文本特征提取器,其特征提取能力远远大于 RNN 和 CNN 模型,这也是 BERT 模型的核心优势所在。

Transformer 是一个完全依赖自注意力来计算输入和输出的表示,而不使用序列对齐的递归神经网络或卷积神经网络的转换模型。自注意力的计算方法如下:需要从编码器的每个输入向量中创建三个向量,一个 Query 向量、一个 Key 向量和一个 Value 向量。这些向量是通过将词嵌入向量与 3 个训练后的矩阵 $\boldsymbol{W}_q$ 、$\boldsymbol{W}_k$ 、$\boldsymbol{W}_v$ 相乘得到的,维度默认为 64。为了便于计算,将三个向量分别合并成矩阵,得到自注意力层的计算公式:

$$Z = \text{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \text{softmax}(\boldsymbol{Q}\boldsymbol{K}^{\text{T}}/\sqrt{d_k})\,V \tag{1}$$

其中, $\boldsymbol{Q}$ 是 Query 向量组合的矩阵, $\boldsymbol{K}$ 和 $\boldsymbol{V}$ 是由 Key 向量和 Value 向量组合的矩阵, d 表示 Query 向量的维度,除以 $\sqrt{d_k}$ 可以使训练过程中的梯度下降更加稳定。最后由 Softmax 将分数进行归一化,每个单词的得分,决定了对某个位置上的单词进行编码时对其他单词的关注程度。由于 BERT 的目标是生成语言模型,只需要使用 Transformer 的编码器机制。

在 Transformer 的基础上,BERT 使用 Masked LM 来进行无监督预训练。一个深度双向模型,要比单向的“左-右”模型,或者浅层融合“左-右”和“右-左”的模型更高效。为了解决双向训练中每个词在多次上下文可以间接看见自己的问题,BERT 采用随机遮掩一定百分比的输入 token,然后通过预测被遮掩的 token 进行训练。

1.2　单通道 TextCNN 模型

TextCNN 使用双通道,引入通道的目的是希望防止过拟合,可以在不同的通道中使用不同方式的词向量嵌入方式,达到在小数据集合获得比单通道更好的性能。其实直接使用正则化效果更好,该文使用单通道的 TextCNN 模型,其结构如图 2 所示。

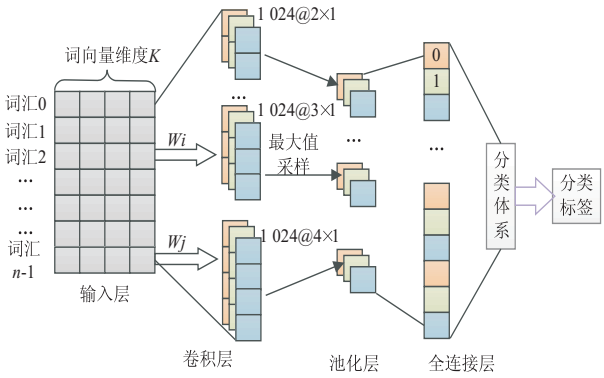

图 2　TextCNN 模型结构

整个模型由四部分构成:输入层、卷积层、池化层、全连接层。TextCNN 模型的输入层需要输入一个定长的文本序列,通过分析语料集样本指定一个输入序列的长度 L ,比 L 短的样本序列需要填充,比 L 长的序列需要截取。对于词向量的表示使用预训练好的 word2vec 作为输入。

在自然语言处理领域,因为在词向量上滑动提取特征没有意义,所以每个卷积核在整个句子长度上进行一维滑动,即卷积核的宽度与词向量的维度等宽,高度与步长可以自定义。通常,在 TextCNN 模型中使用多个不同尺寸的卷积核。卷积核的高度,可以理解为局部词序的长度,窗口值是需要设置的超参数,一般选取 2 ~ 6 之间的值。

在卷积层保留了特征的位置信息,为了保证特征的位置信息在池化层不被丢失,TexCNN 模型选用 k-max pooling 池化方法。相比于最大池化方法,k-max pooling 针对每个卷积核都保留前 k 个最大值,并且保留这些值出现的顺序,即按照文本中的位置顺序来排列这 k 个最大值,对于文本分类精度提升有很大作用。卷积层与池化层的核心作用就是特征提取,从定长文本序列中利用局部词序信息,提取初级的特征,并组合初级的特征为高级特征。

1.3　Bi-GRU 层

GRU 单元保持了 LSTM 的效果,同时又使结构更加简单。GRU 只剩下更新门和重置门两个门限。更新门用于控制前一时刻的状态信息被带入到当前状态

的程度,更新门的值越大说明前一时刻的状态信息带入越多。重置门用于控制忽略前一时刻的状态信息的程度,值越小说明忽略得越多。GRU 单元结构如图 3 所示,GRU 单元的计算公式为:

$$z_{(t)} = \sigma(\boldsymbol{W}_{xz}^{\mathrm{T}} \cdot x_{(t)} + \boldsymbol{W}_{hz}^{\mathrm{T}} \cdot h_{(t-1)}) \tag{2}$$

$$r_{(t)} = \sigma(\boldsymbol{W}_{xr}^{\mathrm{T}} \cdot x_{(t)} + \boldsymbol{W}_{hr}^{\mathrm{T}} \cdot h_{(t-1)}) \tag{3}$$

$$g_{(t)} = \tanh(\boldsymbol{W}_{xg}^{\mathrm{T}} \cdot x_{(t)} + \boldsymbol{W}_{hg}^{\mathrm{T}} \cdot (r_{(t)} \otimes h_{(t-1)})) \tag{4}$$

$$h_{(t)} = (1 - z_{(t)}) \otimes \tanh(\boldsymbol{W}_{xg}^{\mathrm{T}} \cdot h_{(t-1)} + z_{(t)} \otimes g_{(t)}) \tag{5}$$

其中,$\boldsymbol{W}_{xz}$、$\boldsymbol{W}_{xr}$、$\boldsymbol{W}_{xg}$ 是每一层连接到输入向量 x_g 的权重矩阵,$\boldsymbol{W}_{hz}$、$\boldsymbol{W}_{hr}$、$\boldsymbol{W}_{hg}$ 是每一层连接到前一个短期状态 $h_{(t-1)}$ 的权重矩阵。

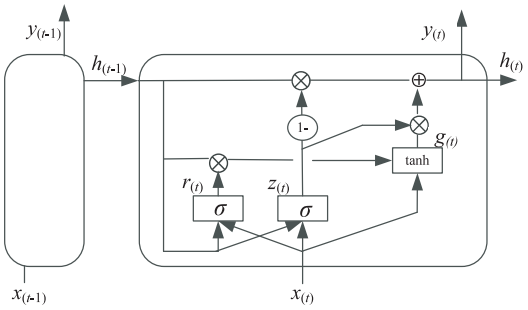

图 3 GRU 单元结构

在处理文本分类问题时,神经网络模型不仅要关注上文信息,同样也要关注下文信息,将前向 GRU 和后向 GRU 结合起来,使得每一个训练序列向前和向后分别是两个循环神经网络,而且这两个网络连接着同一个输出层,这便是 Bi-GRU 的优点。

1.4 多标签分类概率融合

针对训练集,使用 5 折交叉验证方法,首先将训练数据随机分割成 5 个不同的子集,每个子集称为一个折叠。使用第一层文本分类模型对数据进行 5 次训练和评估,每次使用 4 个折叠进行训练,使用另外一个折叠进行评估,评估的结果为每个类别的多标签概率值,而不是分类结果。目前的堆叠模型,初级学习器都是输出分类结果,让混合器在此数据上进行投票,多标签概率值数据远比分类结果值包含更加丰富的信息。

基于标签 m 的输入序列 s_m,假设经 BERT 模型的输出为 z,则经过 Sigmod 函数计算后,标签 m 经第 l 个分类器判定属于分类 q 的概率值 $p_{m,q}^{l}(z \mid s_m)$ 表示为:

$$p_{m,q}^{l}(z \mid s_m) = 1/(1 + \mathrm{e}^{-z}) \tag{6}$$

模型训练完成以后,在验证集上输出每种标签的概率值。以上过程重复 5 次,在交叉验证集上生成一组新的特征数据集。从而可获得任意文本 m 经第 l 个分类器(当 $l=1$ 时,约定分类器为 BERT 模型)分类后的多标签概率值,表示为 $< s_m; p_{m,1}^{1}, p_{m,2}^{1}, \cdots, p_{m,q}^{1}; x_1, x_2, x_3, \cdots, x_Q >$。

同理,另外两个模型的输出结果为:$\langle s_m; p_{m,1}^{2}, p_{m,2}^{2}, \cdots, p_{m,q}^{2}; x_1, x_2, \cdots, x_Q \rangle$ 和 $\langle s_m; p_{m,1}^{3}, p_{m,2}^{3}, \cdots, p_{m,q}^{3}; x_1, x_2, \cdots, x_Q \rangle$。

将多个交叉验证产生的多标签概率值进行融合,对于单个样本来说,相当于将该样本产生的三个多标签概率值向量进行拼接,拼接后的数据作为下一层分类器的输入数据,标记仍然使用原来的 label。对任意输入样本 m,新数据标签集表示为:$\langle s_m; P_m; x_1, x_2, \cdots, x_Q \rangle$。

其中,P_m 为经过 3 个分类器的概率联合,表示为:

$$P_m = \begin{bmatrix} p_{m,1}^{1}, p_{m,2}^{1}, \cdots, p_{m,q}^{1} \\ p_{m,1}^{2}, p_{m,2}^{2}, \cdots, p_{m,q}^{2} \\ p_{m,1}^{3}, p_{m,2}^{3}, \cdots, p_{m,q}^{3} \end{bmatrix} \tag{7}$$

经过 DNN 混合器得到:$P_m \rightarrow L^m$,L^m 表示最终计算出的多标签分类结果。

1.5 DNN 混合器

混合器采用自定义的深度神经网络,输入是基础分类模型计算的联合多标签概率值,输出为样本的多标签分类,其网络结构包含两个隐藏层,每个隐藏层 256 个神经元,采用 He 初始化方法;Dropout 设置为 0.5,使用 ReLU 激活函数。

2 实验结果分析

为了验证所提多标签分类模型的有效性,使用中国裁判文书网公开的裁判文书,以从长文本中抽取案件要素为例,比较该模型与常用模型在分类性能上的差别。

2.1 标注语料

本实验搜集到中国裁判文书网公开的裁判文书 10 万余份。为实现对文本进行分割并分类,需要定义复杂的短文本类别标签集,针对不同的犯罪类型,标签集包含的内容也各不相同。以盗窃罪为例,需要定义的类别有:盗窃时间点、盗窃工具、手段方法、公然窃取、秘密窃取、入户扒窃、造成其他损害、被盗物品价值、失窃者损失后果、处理情况、是否返还、如何到案、强制措施、认罪认罚情况、上诉抗诉等共 15 类标签。由于目前并没有公开的司法文书标注语料库可供使用,因此从语料库中选取盗窃类型且内容较为详实的 2 900 份文书进行标注,所有的标注工作均由经过专业培训的人员手工标注完成。尽管不排除主观因素对多标签标注边界的影响,但总体而言标注质量较高,非常适合用于模型的训练。

标注工作完成后,短文本样本的表示方式为:$\langle s_m, x_1, x_2, \cdots, x_Q \rangle$,其中 s_m 为输入第 m 个的文本序列,x_i 为是否属于标签 i 的示性函数,如果 $x_i = 1$,表示 s_m 属于分类 i,否则 $x_i = 0$。

2.2　样本分布

盗窃案件各要素标签样本分布如表 1 所示。

表 1　盗窃案标签样本分布

标签	训练集	验证集	测试集
盗窃时间点	4 536	1 179	998
盗窃工具	2 237	515	537
手段方法	5 282	1 109	1 532
公然窃取	2 197	615	593
秘密窃取	2 567	642	590
入户扒窃	1 610	419	467
造成其他损害	1 528	321	443
被盗物品价值	6 029	1 447	1 507
失窃者损失后果	671	195	148
处理情况	4 193	1 174	964
是否返还	3 259	880	782
如何到案	1 936	465	465
强制措施	829	199	174
认罪认罚情况	1 365	341	355
上诉抗诉	336	74	94

从数据集中随机抽取三部分作为训练集、验证集、测试集,文书数量比例约为 4 ∶ 1 ∶ 1。第一基础分类模型主要用于获取所有样本的多标签概率分布矩阵,每次使用 80% 的训练集数据进行训练。第二基础分类模型,则使用全部的训练集数据重新训练 3 个第一层分类器,在原来分配的验证集上获取最佳模型。混合器使用多标签概率矩阵进行训练,在训练第二基础分类模型后,使用同一个混合器。两个过程分布完成以后,在最终在测试集上得到整个堆叠模型的性能指标。

2.3　结果对比

在机器学习中评估模型的性能通常使用精度 P 、召回率 R 、F1 分数三个指标,计算公式分别为:

$$P^i = \mathrm{TP}^i/(\mathrm{TP}^i + \mathrm{FP}^i) \tag{8}$$

$$R^i = \mathrm{TP}^i/(\mathrm{TP}^i + \mathrm{FN}^i) \tag{9}$$

$$F^i = 2P^i \times R^i/(P^i + R^i) \tag{10}$$

其中,TP 表示真正类的数量,FP 表示假正类的数量,FN 表示假负类的数量。由公式可知, P 表示精度, R 表示召回率,F1 分数是精度和召回率的谐波平均值,只有当召回率和精度都很高时,才能获得较高的 F1 分数。为了证明提出的模型在性能方面的优越性,在相同数据集上,分别与 TextCNN、BiGRU、BERT 等几个模型进行比较,比较结果如表 2 所示。

表 2　不同模型在测试集上分类性能

标签	TextCNN			Bi-GRU			BERT			堆叠模型		
	P	*R*	F1	*P*	*R*	F1	*P*	*R*	F1	*P*	*R*	F1
盗窃时间点	0.818	0.785	0.801	0.763	0.883	0.819	0.899	0.920	0.909	0.957	0.934	0.946
盗窃工具	0.826	0.760	0.792	0.745	0.805	0.774	0.748	0.903	0.818	0.916	0.832	0.872
手段方法	0.895	0.840	0.867	0.865	0.849	0.857	0.848	0.863	0.856	0.891	0.823	0.855
公然窃取	0.817	0.790	0.803	0.770	0.805	0.787	0.837	0.773	0.804	0.859	0.860	0.860
秘密窃取	0.779	0.769	0.774	0.760	0.776	0.768	0.794	0.822	0.808	0.922	0.843	0.881
入户扒窃	0.808	0.735	0.770	0.785	0.815	0.799	0.754	0.792	0.772	0.806	0.899	0.850
造成其他损害	0.797	0.744	0.770	0.731	0.780	0.755	0.772	0.856	0.812	0.926	0.792	0.854
被盗物品价值	0.893	0.802	0.845	0.872	0.891	0.881	0.866	0.847	0.856	0.895	0.882	0.889
失窃者损失后果	0.772	0.744	0.758	0.763	0.795	0.778	0.805	0.822	0.814	0.847	0.836	0.842
处理情况	0.806	0.775	0.790	0.840	0.892	0.865	0.890	0.853	0.871	0.917	0.879	0.897
是否返还	0.804	0.762	0.782	0.751	0.839	0.793	0.836	0.825	0.831	0.868	0.802	0.834
如何到案	0.823	0.747	0.783	0.767	0.768	0.768	0.835	0.859	0.847	0.863	0.765	0.811
强制措施	0.815	0.719	0.764	0.767	0.781	0.774	0.738	0.780	0.758	0.910	0.734	0.813
认罪认罚情况	0.790	0.713	0.750	0.699	0.763	0.730	0.754	0.855	0.801	0.809	0.862	0.835
上诉抗诉	0.752	0.706	0.728	0.735	0.759	0.747	0.781	0.802	0.791	0.749	0.772	0.760
加权均值	0.832	0.779	0.805	0.798	0.839	0.818	0.836	0.850	0.842	0.894	0.852	0.872

从统计数据可以看出,堆叠模型综合计算 BERT、TextCNN、BiGRU 等强模型输出的分类概率值,在 F1 分数上获得进一步的提升,F1 分数的加权平均值达到 87.2%,比性能最好的 BERT 模型提高了 3 个百分点。

3　结束语

为提高短文本多标签分类性能,提出一种融合深度学习与堆叠模型的短文本多标签分类方法,该方法

采取多层分类器结构,使用 BERT、TextCNN、Bi-GRU 等差异化较大、准确性较高的强分类器作为第一层学习模型,生成的多标签概率矩阵用来训练第二层的混合器。

实验表明,该方法优于目前主流的几种短文本多标签分类模型,在性能上得到了进一步的提升。

参考文献:

[1] AREVIAN G. Recurrent neural networks for robust real-world text classification[C]//Proceedings of the IEEE/WIC/ACM international conference on web intelligence. Rosten, USA: IEEE, 2007: 326-329.

[2] CHEN M, JIN X, SHEN D. Short text classification on improved by learning multigranularity topics[C]//Proceedings of the 22nd international joint conference on artificial intelligence. Lille, France: ACM, 2011: 1776-1781.

[3] WEI F, QIN H, YE S. Empirical study of deep learning for text classification in legal document review[C]//2018 IEEE international conference on big data. Seattle, USA: IEEE, 2018: 3317-3320.

[4] KIM Y. Convolutional neural networks for sentence classification[C]//Proceedings of conference on empirical methods in natural language processing. Lisbon, Portugal: [s. n.], 2014: 1746-1751.

[5] KALCHBRENNER N, GREFENSTETTE E, BLUNSOM P. A convolutional neural network for modelling sentences [C]//Proceedings of the 52nd annual meeting of the association for computational linguistics. Luasanne: ACL, 2014: 655-665.

[6] LEI T, BARZILAY R, JAAKKOLA T. Molding CNNs for text: non-linear, non-consecutive convolutions[J]. Indiana University Mathematics Journal, 2015, 58(3): 1151-1186.

[7] ZHANG Xinwei, WU Bin. Short text classification based on feature extension using The N-Gram model[C]//2015 12th international conference on fuzzy systems and knowledge discovery. Nevada, USA: IEEE, 2015: 710-716.

[8] 陈 钊,徐睿峰,桂 林,等. 结合卷积神经网络和词语情感序列特征的中文情感分析[J]. 中文信息学报, 2015, 29(6): 172-178.

[9] SHI B, BAI X, YAO C. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(11): 2298-2304.

[10] 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. New York, USA: ACM, 2017: 6000-6010.

[11] YANG Z C. Hierarchical attention networks for document classification[C]//Conference of the North American chapter of the association for computational linguistics: human language technologies. Vancouver, Canada: ACL, 2017: 1480-1489.

[12] XIAO Y, CHO K. Efficient character-level document classification by combining convolution and recurrent layers[J]. Computation and Language, 2016, 20(2): 12-20.

[13] HASSAN A, MAHMOOD A. Convolutional recurrent deep learning model for sentence classification[J]. IEEE Access, 2018, 6(1): 13949-13957.

[14] YIN Wenpeng, HINRICH S. Attentive convolution: equipping CNNs with RNN-style attention mechanisms[J]. Transactions of the Association for Computational Linguistics, 2018, 6(1): 687-702.

[15] 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 association for computational linguistics. Florence, Italy: ACL, 2019: 4171-4186.