基于深度学习的 MOOC 作弊行为检测研究

万子云[1], 陈世伟[2], 秦 斌[3], 聂 伟[1], 徐 明[3]

[1]深圳大学电子与信息工程学院 深圳 中国 518061
[2]深圳大学机电与控制工程学院 深圳 中国 518061
[3]深圳大学信息中心 深圳 中国 518061

摘要 快速准确地检测出 MOOC 学习者的作弊行为, 对维护 MOOC 平台的发展及学习者的正常学习具有重要意义。本文研究了一种深度学习混合模型用于 MOOC 作弊行为的检测。该模型通过融合了卷积神经网络、双向门控循环单元以及注意力机制, 大大提升了单一模型的检测性能。本文选取某 MOOC 平台的学习行为数据进行了实验验证, 实验结果显示该模型在验证集上的精确率、召回率、AUC 和误报率分别达到 98.51%、81.35%、91.07%和 0.016%, 具有良好的应用前景。另外, 本文采用了数据扩增的方法以解决 MOOC 作弊行为检测中存在的数据不均衡问题, 实验中通过该方法进行数据平衡后, 该模型在相同的验证集上的 AUC 提升了 1.78%。

关键词 作弊行为检测; 深度学习; 卷积神经网络; 双向门控循环单元; 注意力机制
中图法分类号 TP183 **DOI 号** 10.19363/J.cnki.cn10-1380/tn.2021.01.03

Research on MOOC Cheating Detection Based on Deep Learning

WAN Ziyun[1], CHEN Shiwei [2], QIN Bin[3], NIE Wei[1], XU Ming[3]

[1]School of Electronics and Information Engineering, Shenzhen University, Shenzhen 518061, China
[2]School of Mechatronics and Control Engineering, Shenzhen University, Shenzhen 518061, China
[3]Information Institute, Shenzhen University, Shenzhen 518061, China

Abstract It is of great significance to detect cheating behaviour of MOOC learners quickly and accurately. In this paper, a hybrid model of deep learning is proposed for MOOC cheating detection. By combining Convolutional Neural Networks (CNN), Bidirectional Gated Recurrent Unit (BiGRU) and Attention mechanism, the model greatly improves the detection performance of a single model. On the data sets of a mooc platform, the experimental results show that the precision rate, recall rate, AUC and false positive rate of the model can reach 98.51%, 81.35%, 91.07% and 0.016% respectively, which have good application prospects. In addition, in order to solve the problem of data imbalance in MOOC cheating detection, the paper adopts the method of data augmentation, and the AUC of this model is improved by 1.78% by this method.

Key words cheating detection; deep learning; convolutional neural networks; bidirectional gated recurrent unit; attention mechanism

1 引言

近年来, 大规模开放在线课程(Massive Open Online Courses, MOOC)作为一种全新的网络在线课程模式受到了广泛的关注, 但是相比于线下教学, 缺少了必要的监督, 导致学习者通过作弊手段来完成相应的学习任务的现象层出不穷, 例如刷课、抄袭、替考等。随着近几年高校对 MOOC 学分认可度的提高, 对 MOOC 的作弊行为进行检测势在必行。

针对 MOOC 存在的作弊问题, 常规的解决方法分为两种, 一种是采用被动防护的手段来阻止学习者作弊或者增加学习者作弊的难度; 另一种则是采用主动的检测技术来发现作弊行为并进行相应的处理, 以减少作弊行为的发生。对于第一种解决方法, 一般是通过技术手段禁止学习者在在线学习过程做一些违规操作。例如, 采用操作系统内核的 API 调用技术、

通讯作者: 秦斌, 硕士, 高级工程师, Email:qinbin@szu.edu.cn。
本课题得到深圳大学和深信服科技股份有限公司广东省联合培养研究生示范基地资助; 深圳大学 2020 年研究生教育改革项目(No.860-000001050503)资助。
收稿日期: 2020-08-12; 修改日期: 2020-11-16; 定稿日期: 2020-11-25

系统消息拦截技术、回调技术、钩子技术、注册表访问技术等手段对一些违规操作实施禁用或者屏蔽, 从而达到禁止学生进行页面切换、答案复制及互助抄袭的目的, 但是这种通过技术的手段来限制作弊的方法往往是不够的, 因为计算机系统庞大复杂, 这种硬编码的防护方法在新的作弊手段下效果欠佳, 所以还是需要采用检测技术来对作弊行为进行检测。

对于 MOOC 的作弊行为检测, 工程应用上一般是通过人工检测与规则检测相结合的方式进行排查, 但这种方法不仅要消耗大量的人力物力, 还存在着检测效率低和检测效果欠佳等问题。学术研究上, 常永虎等人[1]基于考生在网络考试中的行为数据, 提出了一种基于互相抄袭的作弊检测算法, 该算法通过计算考生在答题的时间和答案上的相似度来判断作弊的可能性。Ruiperez-Valiente J A 等人[2]开发了一种算法来识别使用 CAMEO(使用多个账号复制答案)方法进行作弊的学习者, 该算法通过比较学习者、问题和提交特征对 CAMEO 的影响, 建立了一个不依赖 IP 的随机森林分类器, 以识别 CAMEO 学习者。Sangalli V A 等人[3]针对学习者互相分享答案以及使用虚假账号获取正确答案这两种作弊手段, 设计了一些指标来得到相应的特征, 再利用 K-means 聚类算法对其进行聚类来识别使用这两种作弊手段的学习者。上述研究都是针对于某种特定作弊形式, 构造相应的特征后采用统计学或机器学习的方法进行作弊检测。本文研究一种更加通用的, 可解决多种作弊形式的作弊检测模型。

学习者在学习过程中所执行的学习动作路径是不一样的[4], 例如正常的学习者会先观看学习视频, 再做练习, 最后提交答案, 异常的学习者可能会直接做练习或者集中在某一个时间段进行刷课等。从本质上看, MOOC 的作弊行为检测属于一种异常行为检测问题。

异常行为检测问题广泛存在于各个领域, 例如网络入侵检测[5-6]、信用卡欺诈检测[7]、故障检测[8]、居民用电检测[9]。根据已有的文献研究, 异常行为检测算法可以分为基于人工提取特征的传统机器学习算法和基于自动化提取特征的深度学习算法[10], 但是传统机器学习算法过度依赖于人工特征提取, 且常常由于特征提取不完整, 导致模型性能不佳等问题。因此, 许多学者尝试使用基于深度学习的方法进行异常行为检测。

卷积神经网络(Convolutional Neural Networks, CNN)[11]与循环神经网络(Recurrent Neural Networks, RNN)[12]是比较常见的深度学习网络模型, CNN 的优势在于能够在空间维度上提取局部特征, RNN 的优势在于能够在时间维度上提取时序特征。对于复杂的 MOOC 学习者的学习行为数据, 一般是具有序列性质的, 因此, 在挖掘学习行为特征时, 需要考虑其空间上的联系, 也需要考虑其时间维度上的关联信息。本文尝试将深度学习模型应用于 MOOC 作弊行为检测中, 结合 CNN 和 RNN 网络模型来对学习者的学习行为序列进行建模, 但是, 普通的 RNN 一方面存在着梯度消失的问题, 另一方面只能学习单个方向的时序特征, 为了解决这些问题, 本文将 RNN 网络替换成其变种网络—双向门控循环单元(Bidirectional Gated Recurrent Unit, BiGRU)[13], 采用 CNN-BiGRU 联合网络提取学习行为序列的空间及时序特征。注意力(Attention)机制[14]是一种模拟人脑注意力机制的模型, 已有实验表明, 融入注意力机制的循环神经网络比单一的网络在机器翻译、情感分类、异常检测等问题中有更好的表现, Brown A 等人[15]在 RNN 中引入了注意力机制, 明显提高了系统日志异常检测的性能。

本文的贡献包括以下三个方面的内容:

(1) 针对之前的MOOC作弊行为检测方法存在的应用场景单一化, 过度依赖人工提取特征, 检测效果不稳定等问题, 本文提出了一种基于CNN-BiGRU-Attention联合网络的MOOC作弊行为检测模型, 该模型融合了CNN、BiGRU、Attention三层网络结构, 实现了自动化特征提取, 可以适用于多种作弊形式的检测, 性能较好。

(2) 针对实际情况下, MOOC作弊行为检测中存在的数据类别不均衡问题, 采用数据扩增的方法增加少数类样本量后再进行模型训练, 增强了模型的泛化能力。

(3) 实现了原型系统, 并基于真实场景下的数据进行了实验验证, 证明了本文提出方法的有效性。

本文后续章节安排如下: 第二部分, 介绍MOOC原始行为日志数据的分析及处理过程; 第三部分, 详细介绍MOOC作弊行为检测模型设计方案; 第四部分, 对模型的性能进行实验验证分析; 第五部分, 结论。

2 数据处理及分析

MOOC 平台的行为日志文件是学习者学习行为数据记录的主要载体, 每一条行为日志数据详细记录了学习者与 MOOC 平台的交互信息。例如访问时间、访问 IP、访问路径、请求数据、响应数据、访问者信息等。图 1 展示的是某学习者一条完整的行为日志数据, 本文对行为日志数据进行 json 解析, 提取与研究问题相关的数据信息。

{"id":"eea3cfc938ec91852281a7786f6f5599"
,"time":"2019-10-23 03:00:01"
,"host":"www.uooconline.com"
,"method":"GET"
,"ip":"119.137.52.247"
,"path":"/test/ffmpegVideo"
,"agent":"curl/7.29.0"
,"response_code":0
,"request_data":"[]"
,"response_msg":""
,"response_data":""
,"uid":0
,"login_by":0
,"gender":0
,"identify":0
,"token":""
,"@version":"1"
,"@timestamp":"2019-10-23T03:00:02.100Z"
}

图 1　学习者行为日志数据

Figure 1　The learner's behaviour log data

图 2 为经过数据解析后某学习者在一个时间段内的学习行为路径数据, 其中, 不同的 Uid 表示不同的学习者, Time 表示学习者执行该动作的时间, Path 表示动作的类型, Mark 是对动作进行的统一标记, Description 是对动作进行的描述。本文以"教学天"为单位对学习者的学习行为动作进行聚合处理, 得到一系列的具有时间先后顺序的行为序列数据, 如图 3 所示。通过对学习行为序列长度进行分析可知, 正常学习者一天 24 小时内产生的学习行为序列长度一般为几十到几百, 而通过作弊学习的行为序列长度往往很长, 极端作弊者一天产生的行为序列长度最长可达几十万。由于行为序列建模与文本序列建模的原理类似, 可以把学习者一个"教学天"的行为序列数据看作一篇文章, 行为序列中的学习动作看作文章中的单词, 由此可以利用 Word2Vec[16]对行为序列中的每个动作进行向量表示, 得到表征行为序列的特征向量矩阵后, 将其作为深度神经网络模型的输入, 完成最终的检测任务。

Uid	Time	Path	Mark	Description
246	2018/9/17 9:12	/user/login	IN	登录
246	2018/9/17 9:13	/home/learn/Video Learn	VL	视频学习
246	2018/9/17 9:27	/home/Threads/post	HTP	论坛发帖
246	2018/9/17 9:35	/home/learn/add Notes	HLA	添加课堂笔记
246	2018/9/17 9:41	/exam/get Task Paper	GTP	开始答题
246	2018/9/17 9:42	/exam/save	ES	保存答题
246	2018/9/17 9:55	/exam/commit	CMT	提交试卷
246	2018/9/17 9:58	/user/logout	OUT	退出

图 2　学习者行为路径数据

Figure 2　The learner's behaviour path data

Uid	Day time	Path list
369	20180917	[IN,VL,VL,VL,HTP,HLA,GTP,ES,ES,ES,CMT,...]
369	20181001	[IN,HW,HW,VL,VL,HTP,HLA,DT,DT,RPL,LK,...]
1789	20180927	[UL,UL,UL,VL,VL,HLA,AN,DT,RPL,AN,RPL,...]
2678	20180922	[AN,HW,HW,,UL,UL,HTP,HLA,DT,DT,RPL,AN,...]
2903	20181105	[PST,VL,VE,AN,VL,HTP,HLA,AN,DT,RPL,LK,...]

图 3　学习者行为序列数据

Figure 3　The learner's behaviour sequence data

3　MOOC 作弊行为检测模型

本文研究了一种深度学习混合模型 CNN- Bi-GRU-Attention 用于 MOOC 作弊行为的检测, 模型结构如图 4 所示。分为嵌入层、卷积神经网络层、双向门控循环单元层、注意力层、输出层。下文将对各层进行详细描述。

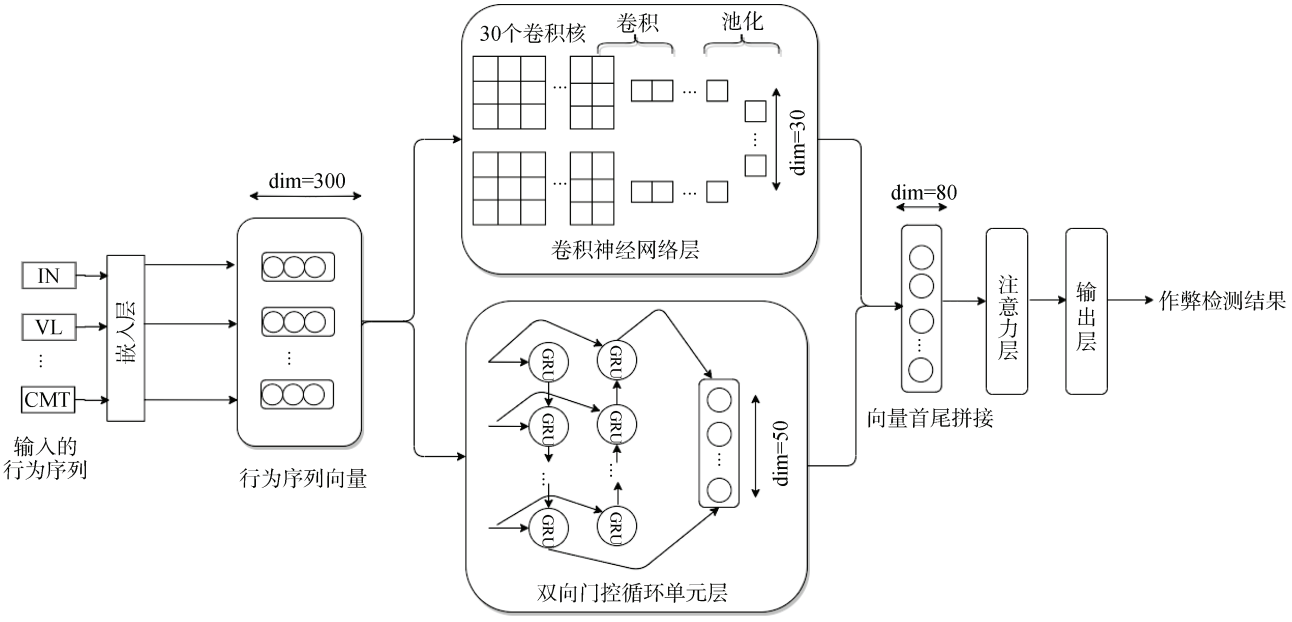

图 4　MOOC 作弊行为检测模型结构

Figure 4　Structure of MOOC cheating detection model

3.1 嵌入层(Embedding)

正如前文所提到的, 先利用 Word2Vec 将行为序列中的行为动作表示成密集的实数向量, 才能将其输入到神经网络模型中, 嵌入层是整个模型的输入, 假设 $x_i \in R^{1*d}$ 表示行为序列中第 i 个动作的词向量, 该词向量的维度为 d, 则长度为 n 的行为序列可表示为实数向量矩阵, 如公式(1)所示。其中, $\oplus$ 符号表示连接操作。

$$x_{1:n} = x_1 \oplus x_2 \oplus x_3 \oplus \cdots \oplus x_n \tag{1}$$

获得行为序列的向量矩阵表征后, 将其作为 CNN 层的输入, 对其进行卷积与池化操作, 提取行为序列中的局部空间特征。同时将其作为 BiGRU 层的输入, 提取行为序列的时序特征。

3.2 卷积神经网络层(CNN)

CNN 最初被广泛应用于计算机视觉领域, 近年来, 随着深度学习在文本领域的研究逐渐增多, CNN 也被应用于文本领域, 用于提取文本序列的局部特征。考虑到学习者学习行为序列中, 相邻的几个行为之间具有关联关系, 因此采用 CNN 挖掘行为序列的局部信息。CNN 一般包含卷积和池化两种操作, 卷积操作以滑窗的方式在不同的地方提取文本序列中的局部信息, 池化操作一般接在卷积操作的后面, 主要作用是减少数据特征维数, 降低计算复杂度。

3.2.1 卷积(Convolution)

本文使用 30 个尺寸为 3 的卷积核提取行为序列中的局部空间信息。假设一个行为序列共有 n 个行为动作, 每个动作由一个 d 维的向量表示, 使用一个尺寸为 h 的卷积核 ω 与 h 个连续行为动作进行卷积操作后得到相应的特征映射 c_i, 卷积操作可由公式(2)表示。

$$c_i = f(\omega \cdot \boldsymbol{x}_{i:i+h-1} + b) \tag{2}$$

其中: $\boldsymbol{x}_{i:i+h-1}$ 是行为序列中第 i 个到第 $i+h-1$ 个连续行为动作组成的子行为序列向量矩阵; b 是偏置项。通过卷积操作后, 可得到特征映射向量 $\boldsymbol{c} = [c_1, c_2, \cdots, c_{n-h+1}]$。

3.2.2 池化(Pooling)

池化就是对卷积之后得到的特征映射向量 $\boldsymbol{c}$ 进行下采样, 求得局部最优解 M_i, 池化分为最大池化和平均池化, 本文模型使用最大池化, 最大池化操作可以由公式(3)表示。

$$M_i = \max(\boldsymbol{c}) = \max(c_1, c_2, \cdots, c_{n-h+1}) \tag{3}$$

由于池化会中断序列结构, 因此将经过池化后的 M_i 连接成特征向量 $\boldsymbol{u}$, 如公式(4)所示。其中, K 表示卷积核的个数。

$$u = (M_1, M_2, \cdots, M_K) \tag{4}$$

3.3 双向门控循环单元层(BiGRU)

RNN 是一类用于处理序列数据的神经网络模型, 通过网络中的循环结构单元记录序列数据的历史信息, 即当前隐藏层的输出不仅与当前时刻的输入有关, 还与前一时刻隐藏层的输出有关[17]。普通 RNN 可以有效地利用近距离的语义特征, 但存在着梯度消失的问题[18], 为了解决该问题, RNN 出现了 LSTM[19],GRU[20]等变体, GRU 其实是 LSTM 的一种改进, 它们都通过“门机制”来记忆前面的序列信息, 以弥补普通 RNN 的不足。不过相比于 LSTM 的三个门单元, GRU 只有两个门单元, 分别为更新门和重置门, 其模型更简单、参数更少, 收敛速度更快。

图 5 为 GRU 单元结构图。z_t 和 r_t 分别为 GRU 的更新门和重置门, x_t 是 t 时刻 GRU 单元的输入, h_t 是 t 时刻 GRU 单元输出的隐藏信息。GRU 单元的具体计算过程如下所示:

$$z_t = \sigma(\omega_z \cdot [h_{t-1}, x_t]) \tag{5}$$

$$r_t = \sigma(\omega_r \cdot [h_{t-1}, x_t]) \tag{6}$$

$$\tilde{h}_t = \tanh(\omega_h \cdot [r_t \times h_{t-1}, x_t]) \tag{7}$$

$$h_t = (1 - z_t) \times h_{t-1} + z_t \times \tilde{h}_t \tag{8}$$

其中, ω_z, ω_r, ω_h 分别为更新门, 重置门以及候选隐含状态的权重矩阵。

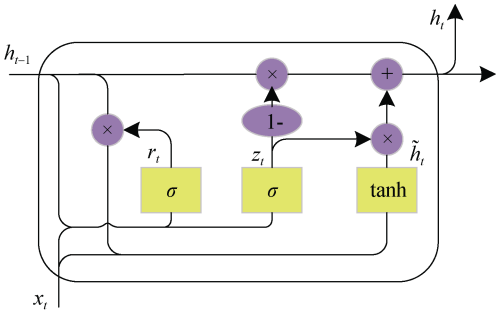

图 5 GRU 单元结构图

Figure 5 GRU unit structure diagram

虽然 GRU 能够很好地捕捉到行为序列的长距离信息, 但是单向的 GRU 在 t 时刻只能捕捉到 t 时刻之前的历史信息, 为了捕捉到前后行为之间完整的关联信息, 本文使用双向的 GRU(BiGRU)网络对行为序列进行建模, BiGRU 既考虑了 t 时刻之前的行为信息, 同时考虑了 t 时刻之后的行为信息。

图 6 为 BiGRU 网络结构图, 从图 6 可以看出, BiGRU 网络包含前向 GRU 和后向 GRU。t 时刻, 前向 GRU 的隐藏状态 $\vec{h}_t$ 由 x_t 和 $\vec{h}_{t-1}$ 决定, 可以获取到

t 时刻之前的行为信息, 后向 GRU 的隐藏状态$\vec{h}_t$由x_t和$\vec{h}_{t-1}$决定, 可以获取到 t 时刻之后的行为信息, 然后再通过向量拼接的方式得到最终的隐藏状态, 这样, 行为序列中的每个行为的隐层状态都包含完整的前后关联信息。相较于单向的GRU而言, BiGRU可以挖掘出更为全面的特征信息。BiGRU 具体的计算过程如式(9)、(10)、(11)所示。

$$\vec{h}_t=GRU(x_t,\vec{h}_{t-1}) \tag{9}$$

$$\overleftarrow{h}_t=GRU(x_t,\overleftarrow{h}_{t-1}) \tag{10}$$

$$h_t=[\vec{h}_t,\overleftarrow{h}_t] \tag{11}$$

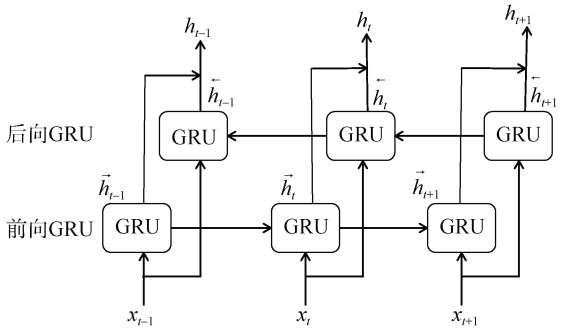

图 6 BiGRU 网络结构图

Figure 6 BiGRU network structure diagram

3.4 注意力层(Attention)

Attention 机制的作用是通过对模型输入特征赋予不同的权重, 加强重要信息对最终结果的影响。在作弊行为检测的过程中, 每个行为特征对最终检测结果的贡献度是不同的, 基于此, 本文引入 Attention 机制来对不同的特征分配不同的权重值。此前已通过 CNN、BiGRU 网络分别获取了行为序列的局部特征向量、时序特征向量, 为了更加完整的表征行为序列, 将局部特征向量和时序特征向量进行首尾相连得到新的行为序列特征向量 z_i, 然后将其输入到 Attention 层得到作弊行为的最终表示。

Attention 层的计算过程如公式(12)、(13)、(14)所示。

$$m_i=\tanh(\omega_1 z_i+b) \tag{12}$$

$$\alpha_i=soft\max(\omega_2 m_i) \tag{13}$$

$$\gamma=\sum_i \alpha_i z_i \tag{14}$$

其中, ω_1和ω_2为权重矩阵; b 为偏置项; γ为注意力层的输出。

3.5 输出层(Output)

输出层实际上是一个 Sigmoid 分类器。经过前面几步, 我们已经得到了行为序列的最终表征向量, 将其输入到 Sigmoid 分类器中进行分类得到作弊检测结果。

4 实验

本文实验环境为一台高性能服务器, 搭载 Centos7 操作系统; CPU 为 Corei5-8300H, 128G 内存; 硬盘配置 2 块 1TB 的 3.5 寸 SATA; GPU 为 QuadroGP100, 16GB 的 HBM2 显卡。本文使用 Keras 来实现模型的搭建, Keras 是一个用 Python 编写的高级神经网络 API, 它能够以 TensorFlow、CNTK 或者 Theano 作为后端运行, 旨在完成深度学习的快速开发。

4.1 数据来源

本文从某 MOOC 在线学习平台的数据库中获取了 210 门学分课程总计 203.5GB 的脱敏数据, 时间跨度为 2018 年的 9 月到 2019 年的 12 月, 涵盖了 60 个院校共 46920 名学习者的学习行为轨迹。对原始数据进行预处理, 最终提取了 1788416 条学习行为序列来验证所提方法的有效性, 选取其中 75%的样本作为训练集, 25%的样本作为验证集。

4.2 参数设置

参数设置会直接影响本文模型的检测效果, 模型的参数设置如表 1 所示。

表 1 模型的参数设置

Table 1 Parameter settings of the model

模型参数	参数说明	参数取值
n_filters	卷积核的个数	30
filter_size	卷积核的大小	3
max_len	输入序列截断长度	2000
vector_length	Word2Vec 嵌入层维度	300
GRU_hidden_dim	GRU 单元中神经元数量	50
Learning_rate	学习率	0.01
epoch	迭代次数	10
Batch_size	批次大小, 每次训练选取的样本数量	128

4.3 实验结果与分析

4.3.1 MOOC 作弊行为检测模型对比

为了评估本文模型的检测性能, 分别选用 CNN、LSTM、GRU、BiGRU、CNN-BiGRU、BiGRU-Attention 等方法在验证集上进行对比实验, 结果如表 2 所示。从实验结果来看, 本文提出的 CNN-BiGRU-Attention 模型取得了最高的精确率、召回率、AUC 和最低的误报率, 分别为 98.51%、81.35%、91.07%和 0.016%, 可见本文模型的检测效

果优于其对比模型。对比前四组实验, 可以看出, 相比于 CNN 和 LSTM 模型, GRU 的作弊检测性能更好, 另外, BiGRU 对比单向的 LSTM 和 GRU, 模型的各个性能指标均有提升, 说明采用双向结构的 BiGRU 能更充分的提取序列的上下文信息, 进而提高了作弊行为的检测能力。对比第一组、第四组和第五组实验, CNN-BiGRU 联合模型相比于单一的 CNN 或者单一的 BiGRU 模型, 在精确率、召回率和 AUC 上都有明显的提升, 主要是因为 CNN-BiGRU 联合模型同时结合了 CNN 和 BiGRU 模型结构的优势, 既学习了行为序列的空间特征, 又学习了行为序列的时序特征。对比第四组和第六组实验, 可以看出, 在 BiGRU 的基础上引入注意力机制后, 检测模型的精确率、召回率和 AUC 值分别提升了 1.15%、0.4%和 1.44%, 究其原因, 主要是因为引入注意力机制后, 对检测贡献度大的特征给予了更高的权重, 提升了重要特征对行为序列分类的影响力。本文提出的 CNN-BiGRU-Attention 网络模型, 由于同时结合了 CNN、BiGRU 以及注意力机制等网络结构的优势, 模型的检测性能进一步得到了提升。

表 2　实验结果

Table 2　Experimental results

方法	精确率(%)	召回率(%)	AUC(%)	误检率(%)	模型推断时间(ms/每个样本)
CNN	90.23	73.56	83.32	0.028	0.721
LSTM	94.87	76.11	86.42	0.021	0.963
GRU	95.23	76.38	86.45	0.019	0.882
BiGRU	96.38	78.56	88.21	0.018	1.759
CNN-BiGRU	97.46	78.83	88.89	0.017	3.422
BiGRU-Attention	97.53	78.96	89.65	0.017	3.526
CNN-BiGRU-Attention	98.51	81.35	91.07	0.016	4.043

此外, 在实验过程中发现, 由于数据规模较大, 上述模型往往在第一个 epoch 就能达到收敛, 为此选取了上述模型中表现较好的三个模型,绘制了它们在第一个 epoch 的训练损失曲线, 如图 7 所示, 可以看出, 相较于 CNN-BiGRU、BiGRU-Attention, CNN-BiGRU-Attention 的收敛速度最慢, 但是损失最低, 模型拟合效果最好。

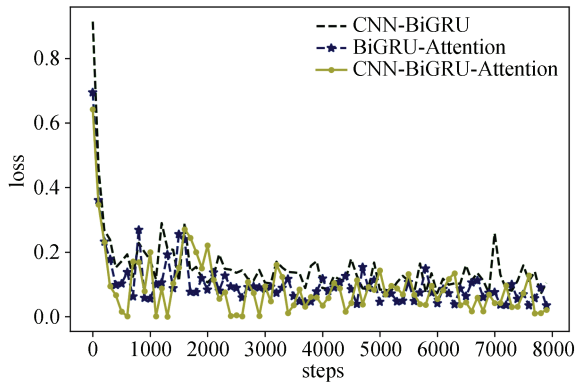

图 7　模型的训练损失曲线

Figure 7　Training loss curves of the models

4.3.2　数据扩增性能提升对比

在训练深度学习模型时, 常常会出现模型过拟合的现象, 而导致这一现象出现的原因很可能就是训练样本不足或者训练样本类别不均衡。在进行 MOOC 作弊行为检测研究时, 相对于正常样本, 作弊样本往往是少之又少。基于此, 本文采用序列截断扩增、平移扩增这两种数据扩增方法来增加作弊行为序列的样本量。截断扩增, 具体而言就是对长度过长的作弊行为序列进行截断, 将截断后的序列打上作弊的标签, 从而增加作弊标签的样本量。而平移扩增是指通过时间滑窗的方式, 以 24 h 的固定窗口前后滑动, 获取一段新的作弊行为序列。实验表明, 通过数据扩增后再进行模型训练, 能够提高模型的泛化能力。经过数据扩增前后的训练集样本分布如下表 3 所示。

表 3　数据扩增前后的训练集样本分布

Table 3　Sample distribution of the training set before and after the data augmentation

	正常样本	作弊样本	样本比例
数据扩增前的样本数量	1326733	14579	约 91∶1
数据扩增后的样本数量	1326733	29483	约 45∶1

图 8 为采用 LSTM, GRU, BiGRU 以及本文模型分别对数据扩增前后的训练集进行模型训练后, 在验证集上得到的 AUC 结果。本文提出的 CNN-BiGRU-Attention 模型在数据扩增前, 在验证集上获取的 AUC 为 91.07%, 而经过数据扩增后, 在相同验

证集上获取的 AUC 为 92.85%, 相比之前提升了 1.78%。LSTM、GRU、BiGRU 经过数据扩增后 AUC 也分别提升了 1.45%, 1.47%, 1.58%。

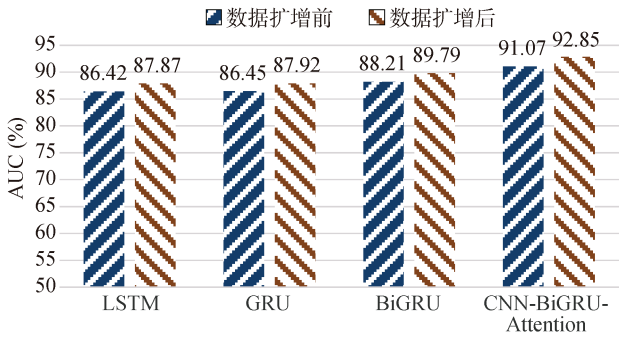

图 8 数据扩增前后的模型 AUC 对比

Figure 8 AUC comparison before and after the data augmentation

5 结论

本文研究了一种基于深度学习混合模型 CNN-BiGRU-Attention 用于 MOOC 作弊行为检测。利用 CNN-BiGRU 联合网络提取行为序列的空间和时序特征, 并引入注意力机制对作弊检测贡献度大的特征给予更高的权重, 然后利用 Sigmoid 分类器进行分类得到作弊检测结果。此外, 通过对比实验 CNN、LSTM、GRU、BiGRU、CNN-BiGRU、BiGRU-Attention 等算法, 证明了本文实验方法的有效性。另外, 针对实际情况下, MOOC 作弊行为检测中存在的数据类别不均衡问题, 采用数据扩增的方法增加少数类样本量, 实验表明, 通过数据扩增后再进行模型训练, 能降低模型过拟合的风险, 提高模型的泛化能力。

本文讨论的是有标签数据集情况下的 MOOC 作弊行为检测, 而在实际情况下, 带有标签的数据是较少的, 后续研究需要进一步结合半监督学习的 MOOC 作弊行为检测方法, 提高作弊检测的准确度。

参考文献

[1] Chang Y H, Luo X, Li H Y. Research on the Mutual Plagiarism Detection Algorithm Based on the Online Behavior of the Examinees[J]. *Journal of Chongqing Technology and Business University (Natural Science Edition)*, 2016, 33(3): 51-55.
(常永虎, 罗旭, 李虎阳. 基于考生在线行为的互抄袭作弊检测算法研究[J]. *重庆工商大学学报(自然科学版)*, 2016, 33(3): 51-55.)

[2] Ruiperez-Valiente J A, Munoz-Merino P J, Alexandron G, et al. Using machine learning to detect 'multiple-account'cheating and analyze the influence of student and problem features[J]. *IEEE Transactions on Learning Technologies*, 2017, 12(1): 112-122.

[3] Sangalli V A, Martinez-Muñoz G, Cañabate E P. Identifying Cheating Users in Online Courses[C]. *2020 IEEE Global Engineering Education Conference*, 2020: 1168-1175.

[4] Pérez-Lemonche Á, Martínez-Muñoz G, Pulido-Cañabate E. Analysing Event Transitions to Discover Student Roles and Predict Grades in MOOCs[M]. Artificial Neural Networks and Machine Learning – ICANN 2017. Cham: Springer International Publishing, 2017: 224-232.

[5] Liu X Q, Shan C, Ren J D, et al. An Intrusion Detection Method Based on Multi-dimensional Optimization of Traffic Anomaly Analysis[J]. *Journal of Cyber Security*, 2019, 4(1): 14-26.
(刘新倩, 单纯, 任家东, 等. 基于流量异常分析多维优化的入侵检测方法[J]. *信息安全学报*, 2019, 4(1): 14-26.)

[6] Jian S J, Lu Z G, Du D, et al. Overview of Network Intrusion Detection Technology[J]. *Journal of Cyber Security*, 2020, 5(4): 96-122.
(蹇诗婕, 卢志刚, 杜丹, 等. 网络入侵检测技术综述[J]. *信息安全学报*, 2020, 5(4): 96-122.)

[7] Dhingra S. Comparative Analysis of algorithms for Credit Card Fraud Detection using Data Mining: A Review[J]. *Journal of Advanced Database Management & Systems*, 2019, 6(2): 12-17.

[8] Riazi M, Zaiane O, Takeuchi T, et al. Detecting the Onset of Machine Failure Using Anomaly Detection Methods[M]. Big Data Analytics and Knowledge Discovery. Cham: Springer International Publishing, 2019: 3-12.

[9] Serrano-Guerrero X, Escrivá-Escrivá G, Luna-Romero S, et al. A Time-Series Treatment Method to Obtain Electrical Consumption Patterns for Anomalies Detection Improvement in Electrical Consumption Profiles[J]. *Energies*, 2020, 13(5): 1046.

[10] Song J M. Analysis of network abnormal behavior based on artifical intelligence[D]. Beijing University of Posts and Telecommunications, 2019.
(宋佳明. 基于人工智能的网络异常行为分析[D]. 北京邮电大学, 2019.)

[11] Gandarias J M, Garcia-Cerezo A J, Gomez-De-gabriel J M. CNN-Based Methods for Object Recognition with High-Resolution Tactile Sensors[J]. *IEEE Sensors Journal*, 2019, 19(16): 6872-6882.

[12] Lu H Y, Jin L, Luo X, et al. RNN for Solving Perturbed Time-Varying Underdetermined Linear System with Double Bound Limits on Residual Errors and State Variables[J]. *IEEE Transactions on Industrial Informatics*, 2019, 15(11): 5931-5942.

[13] Li C, He Y, Li X, et al. BiGRU Network for Human Activity Recognition in High Resolution Range Profile[C]. *2019 International Radar Conference*, 2019: 1-5.

[14] Wang P W, Ji L, Yan J, et al. Concept and Attention-Based CNN for Question Retrieval in Multi-View Learning[J]. *ACM Transactions on Intelligent Systems and Technology*, 2018, 9(4): 1-24.

[15] Brown A, Tuor A, Hutchinson B, et al. Recurrent Neural Network Attention Mechanisms for Interpretable System Log Anomaly Detection[C]. *The First Workshop on Machine Learning for Computing Systems - MLCS'18*, 2018: 1-8.

[16] Dhariyal B, Ravi V. Word2Vec and Evolutionary Computing Driven Hybrid Deep Learning-Based Sentiment Analysis[M]. Advances in Intelligent Systems and Computing. Singapore: Springer Singapore, 2020: 1-16.

[17] Chen R, Ren C G, Wang Z Y, et al. Attention Based CRNN for Text Classification[J]. *Computer Engineering and Design*, 2019, 40(11): 3151-3157.
(陈榕, 任崇广, 王智远, 等. 基于注意力机制的 CRNN 文本分类算法[J]. *计算机工程与设计*, 2019, 40(11): 3151-3157.)

[18] Wang L Y, Liu C H, Cai D B, et al. Chinese Text Sentiment Analysis Based on CNN-BiGRU Network with Attention Mechanism[J]. *Journal of Computer Applications*, 2019, 39(10): 2841-2846.
(王丽亚, 刘昌辉, 蔡敦波, 等. CNN-BiGRU 网络中引入注意力机制的中文文本情感分析[J]. *计算机应用*, 2019, 39(10): 2841-2846.)

[19] Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. *Neural Computation*, 1997, 9(8): 1735-1780.

[20] Dey R, Salemt F M. Gate-variants of gated recurrent unit (GRU) neural networks[C]. *2017 IEEE 60th international midwest symposium on circuits and systems. IEEE*, 2017: 1597-1600

万子云 现于深圳大学电子与通信工程专业攻读硕士学位, 研究领域为大数据与网络空间安全, 研究兴趣领域包括: 大数据技术、人工智能。Email:1732931453@qq.com

陈世伟 硕士, 于 2020 年在深圳大学控制工程专业获得硕士学位。研究领域为数据分析与挖掘, 研究兴趣领域包括: 网络安全、人工智能。Email:2418891877@qq.com

秦斌 副主任, 硕士生导师, 现工作于深圳大学信息中心。研究领域为大数据网络空间安全, 研究兴趣领域包括: 大数据、网络安全、人工智能。Email:qinbin@szu.edu.cn

聂伟 硕士生导师, 现工作于深圳大学电子与信息工程学院。研究领域为软件定义网络, 研究兴趣领域包括: 网络安全、大数据技术。Email:niewei@szu.edu.cn

徐明 硕士生导师, 现工作于深圳大学信息中心。研究领域为教育大数据, 研究兴趣领域包括: 大数据技术、人工智能。Email: xuming@szu.edu.cn