2020 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC) | 978-1-7281-8448-7/20/$31.00 ©2020 IEEE | DOI: 10.1109/CYBERC49757.2020.00039

MOOCs Dropout Prediction Based on Hybrid Deep Neural Network

Yan Zhang[1,2] , Liang Chang[1,2], Tieyuan Liu[1,3]
[1]Guangxi Key Laboratory of Trusted Software
Guilin University of Electronic Technology, Guilin 541004, China.
[2]School of Computer Science and Information Security,
Guilin University of Electronic Technology, Guilin 541004, China.
[3]School of Electronic Engineering and Automation,
Guilin University of Electronic Technology, Guilin 541004, China.
e-mail: 1808305012@mails.guet.edu.cn, changl@guet.edu.cn, 1379268391@qq.com

***Abstract*—The flexibility of Massive Open Online Courses (MOOCS) have attracted more and more learners because it allows students to study at their own pace. However, this flexibility makes it easy for students to drop out of a class. Previous studies, show that the completion rate of MOOC courses was less than 10%. Therefore, a more reliable and efficient dropout prediction model is in need for the development of the MOOC platform. In this paper, our goal is to predict if a learner is going to drop out within the next 10 days, given clickstream data for the first 30 days. We propose a hybrid depth neural network to model and to predict learners' dropout behavior. In order to maintain the time series relationship of the original learning behavior records, we use one-hot encoding to transform the student's behavior record data into a two-dimensional matrix representation, and then extract the local features of the behavior matrix by a new convolutional neural network (CNN) and Squeeze-and-Excitation Networks(SE-Net), and finally extract the potential time series relationship between learning behaviors through a gated recurrent unit (GRU) network to improve the prediction performance. We conducted experiments on KDD Cup 2015 dataset and the results show that our proposed method achieves better performance than baseline methods.**

Keywords- MOOCs; dropout prediction; time series relationship; one-hot encoding; GRU; CNN; SE-Net

I. Introduction

In recent years, with the rapid development of internet technology and educational resources, great changes have taken place in education mode and learning mode. Especially affected by this year's epidemic, the number of online learners has far exceeded that of previous years. Massive open online courses (MOOCs) have become increasingly popular. Many MOOCs platforms have been launched [1]. They have gathered various high-quality curriculum resources from all over the world and promote the development of global education.

These learning platforms span the limitations of time and space and attract more and more learners to participate. They have more freedom to decide what to learn, when, where and how to learn. They can even stop learning completely [2]. However, due to the high flexibility of MOOCs platform, the continuous high dropout rate has always been a challenge for MOOCs platform, which has also become one of the main problems affecting the development of MOOCs [3-4]. According to previous research [5], 91% to 93% of students drop out of class or fail to complete the course. This means that most learners have dropped out before the end of the course, which seriously wastes educational resources and affects the development of MOOCs platform. Therefore, the question of how to reduce the dropout rate, improve the degree of completion of the course, and ensure the rational use of educational resources is very important for the development of this education platform.

To solve this problem, some methods [6-13] have been proposed in recent years. Although these methods have achieved satisfactory results, there are still some problems. In terms of feature extraction: most of the existing studies use feature engineering [14] to complete feature extraction. However, feature engineering is a process of manually extracting features from original data. It is difficult to extract meaningful learner behavior features from low-level clickstream data. Feature engineering requires personnel with expertise in the corresponding field, and the extracted features are subjective. They may pay too much attention to unimportant features, ignoring some important patterns and introducing potential noise data. In terms of model construction: the traditional machine learning model is very sensitive to data interference [15]. Due to the flexibility of MOOCS platform learning, students' behavior data may be different. For unstable machine learning algorithm, the prediction accuracy is low.

In order to overcome the above problems, this study combines feature engineering with CNN automatic feature extraction, which not only maintains the time series relationship between learning behaviors, but also considers the effective local features as much as possible. Then, we propose a new hybrid neural network model that combines convolutional neural network (CNN), Squeeze-and-Excitation Networks (SE-Net) and gated recurrent unit (GRU) to extract learner behavior characteristics and predict dropout trends. A new convolutional neural network (CNN) and Squeeze-and-Excitation Networks (SE-Net) are used to extract the local features of learners. Based on the time series characteristics of clickstream data within MOOCs activity records, gated recurrent unit (GRU) network is used to extract the potential time series relationship between learning behaviors and to predict the dropout trends. To evaluate the effectiveness of the proposed model, we conduct experiments on a data set of kddcup2015 competition. The experimental

978-1-7281-8448-7/20/$31.00 ©2020 IEEE

DOI 10.1109/CyberC49757.2020.00039

results show that this method achieves better performance than the baseline model methods.

II. LITERATURE REVIEW

There are two main categories in the prior studies of MOOC dropout. One regards MOOC's dropout prediction as a binary classification problem and use the traditional machine learning method to predict dropout trends. For example, Kloft et al. [4], Taylor et al. [6], Xing et al. [7], Liang et al. [8] have similar implementation methods in the study of dropout prediction. They extracted several features of a learner from the clickstream logs. The features of several weeks are concatenated to common classification models for dropout prediction. One is to treat MOOCs dropout prediction as a time series classification problem and use sequence labeling perspective to predict. For example, Balakrishnan et al. [9] extracted four different types of features from learner activity logs to train their proposed model. They presented a hybrid model which combined Hidden Markov Models (HMM) and logical regression to predict student retention on a single course. Fei et al. [10] extracted seven features of Coursera courses and five features of edX. By considering the dropout prediction task as sequential labeling, a variant of HMM and two Recurrent Neural Network (RNN) models, were trained to predict dropout.

Few studies used deep neural network models for dropout prediction. For example, Whitehill et al. [11] extracted 37 features from the clickstream data and personal information of learners and add up these features of the previous weeks. Furthermore, logistic regression model and fully connected feed-forward network with five hidden layers were trained to predict the dropout. Wang et al. [12] extracted 186 features from the original records for training the baseline model. By using one-hot encoding, each learner's daily behavior record was transformed into a 24×48 matrix, which was used as the input of the model. They proposed a prediction model by combining the CNN and RNN to predict dropout. Wen et al.[13] extracted seven kinds of click stream data in a day, and then jointed the data into a 30×7 feature matrix as the model input. They proposed a new CNN model to predict dropout.

In recent years, deep learning [16] has shown good performance in solving many problems. Among different types of deep learning networks, the research on convolutional neural networks has emerged in large numbers and achieved good results. CNN is a kind of neural network used to process data with grid structure [16], which can automatically extract the vertical and horizontal local relations between adjacent elements in the grid structure. Squeeze and Excitation networks, referred to as SE-Net [17], is a new image recognition structure announced by Momenta, an autonomous driving company, in 2017. SE-Net improves the performance of the model by modeling the correlation between feature channels and strengthening the important features. GRU is a variant of RNN [18], which is widely used to process time series data [19-20]. GRU can not only overcome the gradient disappears of the traditional RNN model, but also has the advantages of simple structure, high computational efficiency and shorter convergence time.

Therefore, considering the disadvantages of manual feature engineering data extraction and the advantages of using original behavior records to capture the correlation between learners' behavior patterns, we combine CNN, SE-Net and GRU into a new hybrid model to solve the problem of MOOCs dropout prediction.

III. METHODOLOGY

Our goal is to predict whether a learner will continue to learn in the last 10 days based on the click stream data of the first 30 days course. W combine CNN, SE-Net and GRU into a new hybrid model to solve the problem of MOOCs dropout prediction.

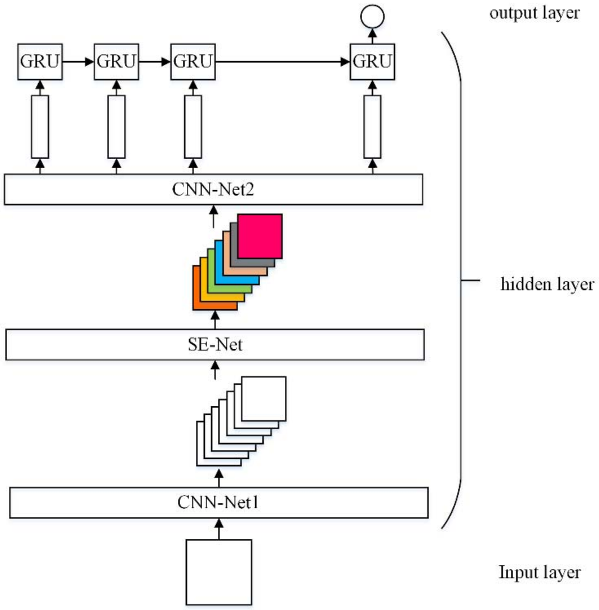


Figure 1. The architecture of our model

The model architecture is shown in Figure 1. Our model includes an input layer, an output layer and a hidden layer module. The hidden layer module is composed of CNN-Net, SE-Net and GRU-Net. The network module is introduced as follows.

A. CNN-Net

In this module, there are two parts of CNN-Net, where CNN-Net1 containing two convolution layers, and CNN-Net2 which contains one convolution layer. In order to maintain the size of input matrix, we use zero padding in the convolution operation, and set "padding" value to be "SAME". In addition, we set the stride to 1 and the convolution kernel size to K×K. Suppose that the output size of an instance in the $(m-1)$ layer is $T^{(m-1)} \times L^{(M-1)}$, after convolution operation of m layer, the output size is $T^{(m)} \times L^{(m)}$, the calculation formula is as follows.

$$\begin{cases} U^{(m)} = \lceil U^{(m-1)}/stride \rceil \\ L^{(m)} = \lceil L^{(m-1)}/stride \rceil \end{cases} \quad (1)$$

The convolution layer uses the rectified liner unit (ReLU) activation function to calculate the output, and the calculation formula is as follows.

$$X^{(m)} = ReLU(W^{(m)}X^{(m-1)} + b^{(m)}) \quad (2)$$

where $X^{(m)}$ represents the output of the m^{th} convolution layer and $X^{(m-1)}$ represents the input of the $(m-1)^{th}$ convolution layer. W^{m} is the filter of the m^{th} convolution layer, which actually is a weight matrix. b^{m} is a bias.

B. SE-Net

The convolution of CNN-Net generated the feature graph U with 64 channels. The weights of each channel of the feature graph are assigned by SE-Net just like the attention mechanism, to help our model learn important feature information. The network structure of SE-Net is shown in Figure 2. Firstly, SE-Net compresses the feature map U along the spatial dimension and transforms each two-dimensional channel into a value representation with global receptive field. Then, the parameters obtained from the compression operation are used to generate weights for each feature dimension through the excitation operation. These weights represent the importance of each feature channel. Finally, the features are weighted to the previous features by multiplication.

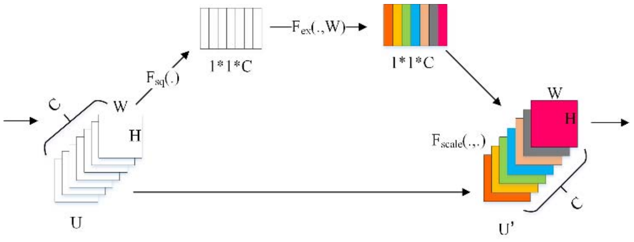


Figure 2. The network structure of SE-Net

- Squeeze operation of SE-Net: SE-Net makes a global average pooling for the output of CNN-Net1, that is, the global spatial information is expressed by the global average pooling into a channel description. As shown in Figure 2, the $F_{sq}(\cdot)$ represents the squeeze process, and the input feature U with the size of $H \times W \times C$ is compressed into the feature description of $1 \times 1 \times C$. For the c^{th} channel, the calculation formula of squeeze operation is as follows.

$$Z_c = F_{sq}(u_c) = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W} u_c(i,j) \quad (3)$$

- Excitation operation of SE-Net: after the above squeeze operation, the network gets a global description, which cannot be used as the weight of the channel. Then through two full connection layers, generate weights for each feature dimension. As shown in Figure 2, the $F_{ex}(\cdot, W)$ represents the process of excitation. The main purpose of this operation is to obtain channel level dependency comprehensively. The first full connection compresses C channels into $\frac{C}{r}$ channels to reduce the computational complexity. In the second full connection, C channels are restored. The full connection layer can fuse all the input feature information well. The calculation formula of exception operation is as follows.

$$s = F_{ex}(z, W) = \sigma\big(g(z, W)\big) = \sigma(W_2\delta(W_1 z)) \quad (4)$$

where $W_1 \epsilon R^{\frac{C}{r}\times C}$ and $W_2 \epsilon R^{C\times\frac{C}{r}}$ are the weight matrices of the two fully connected layers respectively. z is the global description obtained by squeeze operation. δ is the rectified liner unit (ReLU) activation function and σ is the Sigmoid activation function. r is the ratio of compression, which is mainly used to reduce the complexity of network computing and the amount of parameter.

- Fusion operation of SE-Net: after the above exception, the weight of each channel of the feature graph U is obtained. The next step is to fuse the weight and the feature graph U by multiplication as the input data of the next level. As shown in Figure2, the $F_{scale}(\cdot,\cdot)$ represents the squeeze process. The calculation formula of the fusion operation is as follows.

$$U_c' = F_{scale}(u_c, s_c) = s_c * U_c \quad (5)$$

Through the SE-Net network, the important features are enhanced and the unimportant features are weakened, which makes the extracted features more directional.

C. GRU-Net

After SE-Net, GRU-Net is selected to form feature extraction module. Considering the correlation between learners' behavior patterns, this study uses three layers GRU-Net to capture the correlation between behavior patterns in time dimension, and complete the prediction of dropout. GRU-Net is shown in Figure 3.

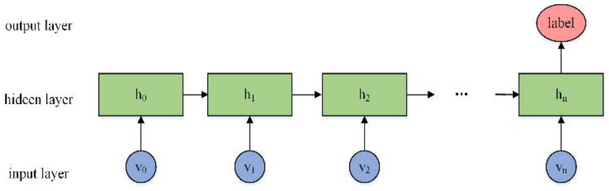


Figure 3. The network structure of GRU-Net

In Figure3, $v_0, v_1, v_2, \cdots, v_n$ represent $n+1$ inputs of GRU-Net and they are the representation of $n+1$vectorization after CNN-Net2. $h_0, h_1, h_2, \cdots, h_n$ corresponds to the hidden layer state at $t_0, t_1, t_2, \cdots, t_4$ respectively. The calculation formula of hidden layer state h_n is as follows.

$$h_n = than(W_1 v_n + W_2 h_{n-1}) \tag{6}$$

where v_n represents a input at time t_n and h_{n-1} is the state of hidden layer at t_{n-1} time. W_1, and W_2 are the weight matrix.

Dropout prediction is a binary classification problem. We use "1" to denote dropout and "0" to indicate no dropout. In the output layer, we use softmax function to predict whether the learner is going to drop out. The calculation formula is as follows.

$$h'_n = W h_n \tag{7}$$

$$y'_i = softmax(h'_n) = \frac{\exp(h'_{n(j)})}{\sum_{j=1}^{T} \exp(h'_{n(j)})} \tag{8}$$

where T is the number of category labels. W is the weight matrix of the model output layer. $h'_{n(i)}$ is the i^{th} component value of the representation vector h'_n .the dimension of vector is equal to the number of categories to be classified. According to the calculation formula of softmax function, we can get a probability distribution. The output value is the value of [0-1] interval, which indicates the possibility of "1".

The loss function of this model uses cross entropy loss function. In the case of two categories, there are only two kinds of results predicted by the model. For each category, we predict the probability of p and $1-p$, and the expression of loss function is as follows.

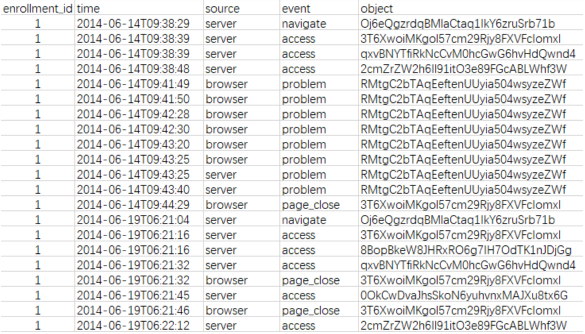
$$L = \frac{1}{N} \sum_i -[y_i \cdot \log(p_i) + ((1-y_i) \cdot \log(1-p_i))] \tag{9}$$

where y_i is the label of sample i. p_i is the probability that sample i is predicted to be positive.

IV. EXPERIMENTAL RESULTS

A. Introduction to the Database

In the study of dropout prediction, the experimental dataset [21] was obtained from 39 courses on the XuetangX platform and it was used in KDD Cup 2015 competition. The dataset includes 39 online courses information, 120542 registration information, 8157277 learning behavior records and the label of whether or not to drop out 10 days after 30 days of learning. On the platform of XuetangX, each learner can register one or more courses. When learning each course, the system will automatically record the clickstream data records of various activity types, as shown in Figure 4. These data are structured, and each record is arranged according to the time sequence of learning behavior. Each behavior record contains different attributes. For example, "enrollment_id" in indicates student registration ID; "time" is the time when the student's corresponding learning behavior event occurs; "source" is the event source of the student's learning behavior, including two kinds; and event is the student's specific behavior event, including 7 kinds; "object" refers to the object that students access or navigate to through access or navigation behavior (specifically refers to a chapter module of a course).



enrollment_id	time	source	event	object
1	2014-06-14T09:38:29	server	navigate	Oj6eQgzrdqBMlaCtaq1IkY6zruSrb71b
1	2014-06-14T09:38:39	server	access	3T6XwoiMKgol57cm29Rjy8FXVFclomxl
1	2014-06-14T09:38:39	server	access	qxvBNYTfiRkNcCvM0hcGwG6hvHdQwnd4
1	2014-06-14T09:38:48	server	access	2cmZrZW2h6Il91itO3e89FGcABLWhf3W
1	2014-06-14T09:41:49	browser	problem	RMtgC2bTAqEeftenUUyia504wsyzeZWf
1	2014-06-14T09:41:50	browser	problem	RMtgC2bTAqEeftenUUyia504wsyzeZWf
1	2014-06-14T09:42:28	browser	problem	RMtgC2bTAqEeftenUUyia504wsyzeZWf
1	2014-06-14T09:42:30	browser	problem	RMtgC2bTAqEeftenUUyia504wsyzeZWf
1	2014-06-14T09:43:20	browser	problem	RMtgC2bTAqEeftenUUyia504wsyzeZWf
1	2014-06-14T09:43:25	browser	problem	RMtgC2bTAqEeftenUUyia504wsyzeZWf
1	2014-06-14T09:43:25	server	problem	RMtgC2bTAqEeftenUUyia504wsyzeZWf
1	2014-06-14T09:43:40	server	problem	RMtgC2bTAqEeftenUUyia504wsyzeZWf
1	2014-06-14T09:44:29	browser	page_close	3T6XwoiMKgol57cm29Rjy8FXVFclomxl
1	2014-06-19T06:21:04	server	navigate	Oj6eQgzrdqBMlaCtaq1IkY6zruSrb71b
1	2014-06-19T06:21:16	server	access	3T6XwoiMKgol57cm29Rjy8FXVFclomxl
1	2014-06-19T06:21:16	server	access	8BopBkeW8JHRxRO6g7IH7OdTK1nJDjGg
1	2014-06-19T06:21:32	server	access	qxvBNYTfiRkNcCvM0hcGwG6hvHdQwnd4
1	2014-06-19T06:21:32	browser	page_close	3T6XwoiMKgol57cm29Rjy8FXVFclomxl
1	2014-06-19T06:21:45	server	access	0OkCwDvaJhsSkoN6yuhvnxMAJXu8tx6G
1	2014-06-19T06:21:46	browser	page_close	3T6XwoiMKgol57cm29Rjy8FXVFclomxl
1	2014-06-19T06:22:12	server	access	2cmZrZW2h6Il91itO3e89FGcABLWhf3W

Figure 4. Clickstream logs

B. Data Processing

The original learning behavior records in dataset are in text format, which can not be directly used as the input of our model. In order to meet the requirements of model input, the data in text format needs to be transformed into the format that can be processed by deep neural network.

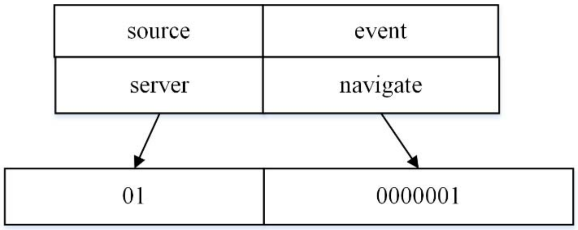


Figure 5. One-hot coding of raw data

First, in this article, we use one-hot encoding [22] to transform each behavior record into a vector. Considering the effectiveness of the data used for dropout prediction, this paper selects two attributes related to the behavior record: source and event. Then, one-hot coding is used to transform it into an effective learning behavior representation vector, as shown in Figure 5. Considering the sparsity problem of generating eigenmatrix, we add the corresponding positions of the representation vectors of all behavior records in a day to generate the behavior representation vector of one day. For an example, zero vector is used to represent a day with any behavior records.

Secondly, considering that some important information may be ignored by adding the representation vectors of all the behavior records in a day, we make a statistical analysis of the learners' learning behavior records. As can be seen in Figure

6, with the increase of students' effective learning time, the dropout rate tends to decrease; As can be seen in Figure 7, with the increase in the number of courses selected, the dropout rate is decreases. As can be seen in Figure 8, for different courses, the dropout rate is different. That is because different courses have different degrees of difficulty, which will also affect the occurrence of students' dropout behavior. Regarding the total number of dropouts, if a learner has dropped out of more than one course, she or he is more likely to give up the current course. Therefore, we take these four attributes into account to form a daily behavior representation vector of students with a size of 1 × 13.

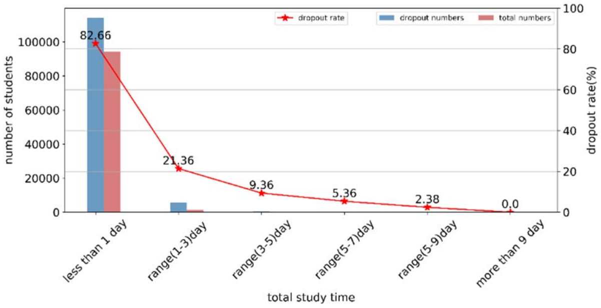


Figure 6. Dropout rates of different effective learning time

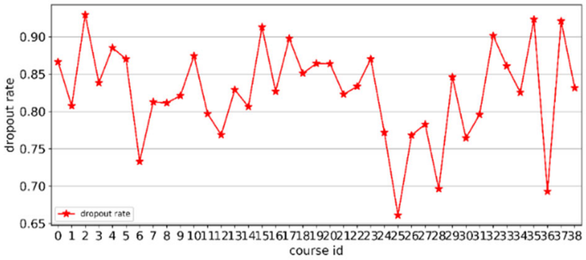


Figure 7. Dropout rates of different course_id

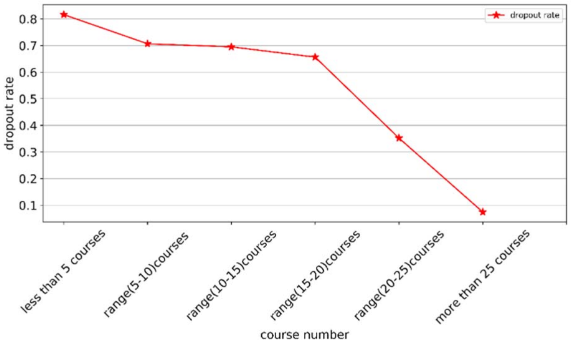


Figure 8. Dropout rates of different course number

Finally, the daily behavior representation vector is jointed into a 30 × 13 behavior representation matrix in order, as shown in Figure 9. In addition, we use (0-1) standardization to process the data in columns as the input of the model.

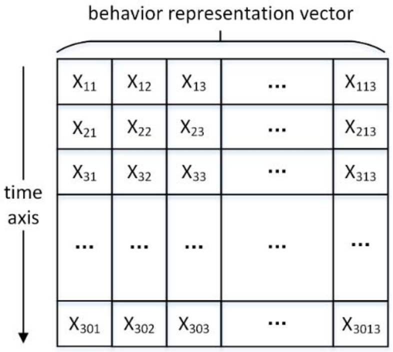

behavior representation vector
time axis
X_{11} X_{12} X_{13} … X_{113}
X_{21} X_{22} X_{23} … X_{213}
X_{31} X_{32} X_{33} … X_{313}
… … … … …
X_{301} X_{302} X_{303} … X_{3013}

Figure 9. Behavior representation matrix

C. *Experimental Settings*

1) Implementation Details: In this experiment, we choose two attributes, "event" and "source", in the original behavior record. In addition, we add three attributes related to learners' behavior patterns: "effective learning time", "number of selected courses" and "course number". Through transformation and splicing, we form a behavior representation matrix with the size of 30×13. There is such a characteristic matrix used to represent the 30 days behavior pattern, also used as the input of the model.We implement our proposed model with TensorFlow and adopt Gradient descent to optimize the model. we adopt Rectified Linear Unit (Relu) as activation function.

2) Comparision Methods: In order to prove the validity of the proposed model, we selected some common classification methods as the baseline, which extracted features by the feature engineering. We refer to the comparison method in [17]. Through the analysis of the original behavior records, we extract seven types of click stream data of events, such as video, problem, discussion, and so on. Then, these features of the extracted 30 days are spliced to form a feature representation vector with a length of 210, which is used to represent the 30 day behavior characteristics of a learner. These features in the training dataset are normalized by min-max scaling. We select the commonly used classification algorithms as baseline methods, which are usually used for dropout prediction, including decision tree (CART), Naive Bayes (NB) , Liner Discriminant Analysis(LDA),Logical Regression(LR), support vector machine (SVM), Gradient Boosting Decision Tree(GBDT), Random Forest(RF). For the setting of baseline model parameters, we also refer to the existing papers [17], as shown in Table I. In addition, two comparative experiments of CNN [17] andCNN-RNN [16] are added.

TABLE I. Parameters setting of baseline model

Method	Parameter	Value
SVM	C,γ	C=1,γ=$\frac{1}{210}$
GBDT	n_estimator	500
RF	n_estimator	500

3) Evaluation Metrics: The dropout prediction problem is actually a binary classification problem. Considering the imbalance of the distribution of positive and negative samples in the experimental data set, we choose Accuracy as the evaluation metric, as well as Precision, Recall and F1-score.

D. Experimental Results and Analysis

The experimental results of our model and baseline method are shown in Table II. Compared with baseline methods, our proposed model achieves better results on four metrics, which also shows the validity and effectiveness of our proposed model.

TABLE II. Performances of different methods on different metrics

Method	Precision	Recall	F1-score	Accuracy
CART	0.8506	0.9666	0.9049	0.8394
NB	0.8800	0.9217	0.9004	0.8388
GBDT	0.8887	0.9615	0.9237	**0.8792**
LDA	0.8583	**0.9768**	0.9137	0.8542
LR	0.8727	0.9668	0.9172	0.8622
RF	**0.8977**	0.9422	0.9194	0.8695
SVM	0.8717	0.9678	0.9172	0.8620
CNN-RNN	0.8862	0.9655	**0.9241**	0.8562
CNN	0.8932	0.9528	0.9224	0.8724
CNN-SE-GRU	**0.9593**	0.9726	**0.9659**	**0.9458**

Previous work has demonstrated the importance of one-hot encoding in the field of Natural Language Processing. In this paper, we use one-hot encoding to code the learner's behavior data and construct the learner's behavior matrix. Compared with traditional machine learning algorithms, for example, CART, NB , LDA, LR, RF, SVM, we find that one-hot encoding can achieve good experimental results in deep learning model, for example, CNN-RNN and CNN-SE-GRU. The traditional machine learning models rely on data extracted by feature engineering in table format. It is very difficult to capture correlation between data. The accuracy of data extraction will affect the performance of the model.

Experimental results demonstrated that the proposed model is better than the baseline model, which can be attributed to two factors. Firstly, the model uses the one-hot encoding to encode the learners' behavior records in the data processing, and CNN can extract local information from the behavior matrix; Secondly, after the convolution of CNN, the model forms multiple feature mapping matrices. SE-Net can weight each mapping matrix to obtain key information. However, the features of baseline model are extracted by feature engineering, and the extracted features are all independent individuals without considering the correlation between students' behavior patterns, so the prediction accuracy is not very high.

V. CONCLUSION

In the problem of MOOC dropout prediction, this study proposes a CNN-SE-GRU hybrid depth neural network to predict the problem of MOOCS dropout. This model can automatically extract important features from the original records, and can effectively avoid the subjectivity and noise data of features extracted by artificial feature engineering. Additionally, SE-Net is used to assign weights to the multi-channel feature matrix after CNN-Net1, and GRU-Net is used to analyze the potential temporal relationship between behavioral data and to predict dropout trends. Compared with previous studies, our model introduces SE-Net after CNN-Net, which can effectively extract features and has higher prediction accuracy. The predicted results can help MOOCS platform teachers to adjust the teaching methods and teaching contents in time, help learners improve the completion of the course, and ensure the rational use of education resources on MOOCS platforms.

In future work, we intend to do further two categories of research: data processing and model building. We will attempt to use additional data sets regarding MOOCs platforms. In data processing, we will attempt to use other encoding methods in the field of natural language processing, such as word embedding. On the premise of more abundant data set information, the model can be further improved by incorporating other optimization algorithms to improve the prediction accuracy.

ACKNOWLEDGMENT

This work is supported by the Natural Science Foundation of China (Nos. U1811264, U1711263, 61966009), the Natural Science Foundation of Guangxi Province (No.2018GXNSFDA281045),and Undergraduate Teaching Reform Project of Guangxi Higher Education (No.2019JGZ118).

REFERENCES

[1] Feng W, Tang J, Liu T X, et al. Understanding Dropouts in MOOCs[C]. national conference on artificial intelligence, 2019, 33(01): 517-524.

[2] Yang, D.; Sinha, T.; Adamson, D.; and Ros´e,C. P. Turn on, tune in, drop out: Anticipating student dropouts in massive open online courses. In NIPS 2013, Data-driveneducation workshop.

[3] Yuan Wang. Exploring Possible Reasons behind Low Student Retention Rates of Massive Online Open Courses: A Comparative Case Study from s Social Cognitive Perspective". In Proceedings of the 1stWorkshop on Massive Open Online Courses at the 16th Annual Conference on Artificial Intelligence in Education, 2013.

[4] M. Kloft, F. Stiehler, Z. Zheng, and N. Pinkwart, Predicting MOOC dropout over weeks using machine learning methods, in Proc. the EMNLP 2014 Workshop on Analysis of Large Scale Social Interaction in MOOCs, Doha, Qatar, 2014, pp. 60–65..

[5] MOOCs on the Move: How Coursera Is Disrupting the Traditional Classroom (text and video). Knowledge@ Wharton. University of Pennsylvania. 7 November 2012. Retrieved 23 April 2013.

[6] C. Taylor, K. Veeramachaneni, and U. O'Reilly, Likely to stop? Predicting stop out in massive open online courses, https://arxiv.org/abs/1408.3382?context=cs.CY, 2014.

[7] W. Xing , Chen , X. Stein J , et al. Temporal predication of dropouts in MOOCs: Reaching the low hanging fruit through stacking generalization[J]. Computers in Human Behavior, 2016, 58(May):119-129.

[8] J. Liang, J. Yang, Y. Wu, C. Li, and L. Zheng, Big data application in education: Dropout prediction in edx MOOCs, in Proc. IEEE Second International Conference on Multimedia Big Data, Taipei, China, 2016, pp. 440–443.

[9] G. Balakrishnan, G. K. Balakrishnan, and D. Coetzee, Predicting student retention in massive open online courses using hidden markov models, Technical Report No. UCB/EECS-2013-109, Electrical Engineering and Computer Sciences University of California, Berkeley, CA, USA, 2013.

[10] M. Fei and D. Yeung, Temporal models for predicting student dropout in massive open online courses, in Proc. IEEE International Conference on Data Mining Workshop, Atlantic City, NJ, USA, 2015, pp. 256–263.

[11] J. Whitehill, K. Mohan, D. Seaton, Y. Rosen, and D. Tingley, Delving deeper into MOOC student dropout prediction, https://arxiv.org/pdf/1702.06404.pdf, 2017

[12] W. Wang, H. Yu, and C. Miao, Deep model for dropout prediction in MOOCs, in Proc. the 2nd International Conference on Crowd Science and Engineering, Beijing, China, 2017, pp. 26–32.

[13] Wen Y , Tian Y , Wen B , et al. Consideration of the local correlation of learning behaviors to predict dropouts from MOOCs[J]. Tsinghua Science and Technology, 2019.

[14] Veeramachaneni K, Oreilly U, Taylor C A, et al. Towards Feature Engineering at Scale for Data from Massive Open Online Courses.[J]. arXiv: Computers and Society, 2014.

[15] ZhiHua Zhou. Ensemble Methods: Foundations and Algorithms[M]. Taylor & Francis, 2012.

[16] Lecun Y , Bengio Y , Hinton G . Deep learning[J]. Nature, 2015, 521(7553):436.

[17] Hu J , Shen L , Albanie S , et al. Squeeze-and-Excitation Networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, PP(99).

[18] Bahdanau D , Cho K , Bengio Y . Neural Machine Translation by Jointly Learning to Align and Translate[J]. Computer ence, 2014.

[19] Yu S , Liu D , Zhu W , et al. Attention-based LSTM, GRU and CNN for short text classification[J]. Journal of Intelligent and Fuzzy Systems, 2020:1-8.

[20] Zhang Y , Lu W , Ou W , et al. Chinese medical question answer selection via hybrid models based on CNN and GRU[J]. Multimedia Tools and Applications, 2019.

[21] Dataset https://data-mining.philippe-fournier-viger.com/the-kddcup-2015-dataset-download-link/

[22] Johnson R, Zhang T. Effective Use of Word Order for Text Categorization with Convolutional Neural Networks[C]. north american chapter of the association for computational linguistics, 2015: 103-112.