Multi-Model Stacking Ensemble Learning for Dropout Prediction in MOOCs

Ya Zhou¹ and Zhixiang Xu¹*
¹ Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, City, Guilin, 541004, China
*Corresponding author’s e-mail: moxi0624@163.com

Abstract. In recent years, with the rapid development of streaming media technology, massive open online courses (MOOC) have attracted unprecedented attention. Compared with the traditional offline teaching mode, the MOOC of online teaching has a higher degree of openness. Users can arbitrarily interrupt the study of a course according to their own interests, and have a higher dropout rate. The existing single machine learning model needs to be improved in terms of prediction accuracy. In this paper, a dropout prediction model based on multi-model stacking ensemble learning (MMSE) is proposed to further improve the accuracy of MOOC dropout prediction. The model mainly includes two parts: data preprocessing and model building. In the data preprocessing part, the student's log records are used to design data features in weeks. In the model building part, a two-layer ensemble learning model is established. In the first layer, 5-fold cross-validation is used to train five different base classifiers. The second layer uses the XGBoost algorithm to combine the prediction results of the first layer to predict MOOC dropout. The experimental results on the KDDcup2015 real data set show that the MMSE model has achieved better results than the single model.

1. Introduction

Nowadays, online education platforms domestic and international, represented by Coursera, Udacity, edX and XuetangX, have developed vigorously. MOOC, a new learning method, has attracted nearly 10 million active participants worldwide. The MOOC platform has the characteristics of openness and online, and students can study courses according to their own interests and hobbies. Compared with the traditional classroom model, due to the lack of real-time interaction and communication between teachers and students, the dropout rate reached 90%. The high dropout rate of MOOC makes the scale of users decline and the platform revenue decreases, becoming the bottleneck of platform development. By analyzing the learner's learning behavior log records, predicting potential dropout students in advance, and taking intervention measures to them, is of great significance to improve students' enthusiasm for learning, thereby reducing the dropout rate. In addition, many educational institutions are currently using accurate dropout predictions to improve course content and teaching quality. Through personalized course recommendation and learning guidance, the dropout rate of students is reduced, thereby increasing the income of the MOOC platform.

To this end, this paper proposes a MMSE dropout prediction model. First, using feature engineering technology to build forty-six features from the student's learning behavior log records in weeks, and at the same time combine four weeks of data to obtain one hundred and eighty-four dimensional features, which greatly enriches the feature dimensions. Secondly, five different machine learning algorithms, AdaBoost, GBDT, XGBoost, LightGBM and CatBoost, are used as base classifiers to predict student
dropouts respectively. Then, the prediction of each base classifier is taken as the input feature of the second layer XGBoost classifier, and the output of the base classifier is fused as the final prediction result of the model.

2. Related work
At present, the research methods of MOOC dropout prediction mainly include traditional machine learning, ensemble learning and deep learning.

Some early researchers tried to use traditional machine learning classification methods to build predictive models, such as Logistic Regression (LR) and Support Vector Machine (SVM) [1]. Taylor et al. selected fourteen weeks of learning behavior data from the learner activity log and used machine learning methods such as LR, SVM, Decision Tree to predict dropout [1]. Because traditional machine learning prediction models have the problem of low accuracy when facing increasing data scales, some scholars try to use ensemble learning methods to improve accuracy. Liang et al. built a Gradient Boost Decision Tree (GBDT) model through registration features, user features and behavioral features to predict whether students will participate in course learning in the next ten days [2]. Lu et al. extracted nineteen features from students’ learning behavior data and used a variety of machine learning algorithms to build a sliding window model to predict students’ dropout rate over a certain period of time [3]. Hong et al. proposed a two-level cascaded prediction model combining Random Forest (RF), SVM and Multinomial Logistic Regression (MLR) based on the unreliable prediction results of a single classifier [4].

With the continuous development of deep learning, the dropout prediction model based on deep learning has also attracted the attention of many scholars. Reference [5] proposes a combination model of Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) that can automatically extract features from the original MOOC dataset in response to the problem of excessive dependence on artificially constructed features and poor model scalability in the existing methods.

3. Data preprocessing and feature engineering
The main purpose of this article is to build an effective dropout prediction model by analyzing students’ learning behavior logs (such as watching videos, doing homework, forum postings, and browsing forums, etc.) to predict whether students will have dropout behavior in the future time period. The data set is derived from 8.15 million student log records of part of the 39 courses in the half-year period of 120,000 xuetangx in the KDD Cup2015 competitions. In order to study the internal correlation between data, we preprocess the data through data cleaning and feature engineering to generate label data.

3.1. Data analysis
In this part, we conducted a statistical analysis of the data set. From the student log behavior records shown in Figure 1, the curve of the number of days of the course and the number of learners found that most learners mainly concentrated on the first Week, then the number of learners will show a cliff-like decline, and finally tend to be flat.

Figure 1. Change curve of learning number.
Looking at the number of dropouts through the students’ dropout labels, it is found that there is an imbalance between the positive and negative sample ratios, that is, the number of dropout users is much higher than that of non-dropout users, and the ratio is about 4:1, as shown in Figure 2.

![Figure 2. Number of dropouts and non-dropouts.](image1)

Randomly select a dropout user and non-dropout user from the pre-processed features, and check their weekly event distribution. As shown in Figure 3, it is found that the learning curve of non-dropout users is relatively smooth, with weekly learning Log records, while dropout users only record in a certain period of time, and the number of events is also lower than non-dropout users.

![Figure 3. Comparison of the number of dropout and non-dropout events.](image2)

3.2. Feature construction
In data mining problems, data characteristics will have a great influence on the final prediction results. The original student learning log data set contains only: the registration number, the time when the event occurred, the type of event, the resource for the event operation, and the object to read or browse. The features of the above five dimensions are directly input into the dropout model for extremely poor training. Therefore, it is necessary to use feature engineering to process the student log records, so that the training data set and the test data set can be input into the machine learning algorithm.

We have constructed forty-six features that contain information about learning behavior in a course for a student from the student registration data set, learning log data set, course module data set, and course start and end time data sets. The specific features are described in Table 1. Show.

| feature | description |
|---------|-------------|
| x1      | Total events in week i |
| x2      | Total activities in week i |
| x3 - x9 | The total number of 7 different types of activities in week i |
| x10     | Total number of browser events in week i |
| x11     | Total number of server events in week i |
| x12     | Active days in week i |
x13 Sessions in week i
x14 The total learning time of week i
x15 - x22 The average of the total of 7 different types of Activities in week i
x23 Number of events using browser in week i
x24 - x27 The average number of events in week i using the browser to do homework, watch videos, access objects, and close the page
x28 Number of events using server in week i
x29 - x33 The average number of events using the server to do homework, access objects, view Wikipedia, participate in discussions, and navigate courses in week i
x34 - x35 Number of browser and server events in week i
x36 - x39 Week i use the browser to do homework, watch videos, access objects, and close the page
x40 - x44 Week i use the server to do homework, access objects, view Wikipedia, participate in discussions, and navigate the number of events in the course
x45 - x46 Average days of using browser and server in week i

3.3. Sample construction
It can be seen from the data analysis in Section 3.1 that the learning log time of the course is around four weeks, so it is necessary to integrate the weekly features as the final features of the data. As shown in Figure 4, each row of data contains the following three parts.

- Index: The index part mainly contains the registration number.
- Features: The features section mainly contains data features around the students
- Label: If the user does not study in the next ten days, then it is defined as dropout, otherwise it is marked as non-dropout.

4. Multi-model stacking dropout prediction model
This paper proposes a multi-model stacking integrated learning MOOC dropout prediction model as shown in Figure 5. Mainly divided into two-layer structure, the first layer includes Adaboost, GBDT, XGBoost, LightGBM and CatBoost 5 base classifiers. The output of each base classifier is a one-dimensional feature. The results of the five classifiers are fused and used as the input of the secondary classifier of the second layer XGBoost model. The output of the second layer model is the final prediction result.
In the first layer of model training, first divide the data set into a training set and a data set at a ratio of 8:2, and use the five base classifiers of AdaBoost, GBDT, XG Boost, LightGBM, and CatBoost to train in parallel on the same data set to obtain five basic classifiers \( M_1, M_2, M_3, M_4, M_5 \). Each basic classifier is trained by a 5-fold cross-validation method, that is, the training set is equally divided into \( D_{train}^0, D_{train}^1, D_{train}^2, D_{train}^3, \) and \( D_{train}^4 \), and each base classifier is trained for five iterations.

\[
M_i^k = N_i \left( D_{train} - D_{train}^k \right) \quad (1)
\]

Where \( M_i^k, i \in (0, 4), k \in (0, 4) \) represents the model generated by the \( i \)-th algorithm in the \( k \)-fold cross-validation.

Finally, predict the \( D_{train} \) through the trained classifier.

\[
\hat{Y}_i^k = M_i^k (D_{train}^k) \quad (2)
\]

The predicted result is shown in Equation 3.

\[
\left( \hat{Y}_i \right)^T = \left( \left( \hat{Y}_i^0 \right)^T, \left( \hat{Y}_i^1 \right)^T, \left( \hat{Y}_i^2 \right)^T, \left( \hat{Y}_i^3 \right)^T, \left( \hat{Y}_i^4 \right)^T \right)
\]

\[
= \left( \hat{Y}_i^0, \hat{Y}_i^1, \hat{Y}_i^2, \ldots, \hat{Y}_i^{n-1} \right) \quad (3)
\]

Where \( \hat{Y}_i \) represents the \( i \)-th basic classifier's prediction of \( n \) samples of \( D_{train} \), which will be used as part of the training set of the second layer of models. In addition, each cross-validated base classifier outputs five prediction results in the test set, the average value of which is part of the test set in the second layer.

\[
\hat{Z}_i^k = M_i^k (D_{test}) \quad (4)
\]

\[
\hat{Z}_i = \frac{1}{5} \sum_{k=0}^4 \hat{Z}_i^k = \left( z_0^i, z_1^i, z_2^i, \ldots, z_{n-1}^i \right) \quad (5)
\]

Where \( \hat{Z}_i \) indicates the prediction of the \( m \)-th \( D_{test} \) sample by the \( i \)-th base classifier.

Train the second layer XGBoost model: In the second layer of the dropout prediction model, the results of the first layer model are combined to construct a new training set \( S_{train} \) and test set \( S_{test} \). The data format is \( \left( \left( \hat{Y}_0^j, \hat{Y}_1^j, \hat{Y}_2^j, \hat{Y}_3^j, \hat{Y}_4^j, y_j \right), j = 0, 1, 2, \ldots, n-1 \) the feature information is the prediction result of the first layer model, and the label is the true label of the sample. The final prediction result is shown in Equation 7.

\[
MMSE = XGBoost(S_{train}) \quad (6)
\]

\[
PredSet = MMSE(S_{test}) \quad (7)
\]

5. Experiment

5.1. Dataset and Evaluation index

This experiment uses the KDDCup2015 competition data set [6]. In this article, we believe that if a student did not leave a diary in this course for ten consecutive days, then he was deemed to have dropped out. The training set is composed of 76517 positive samples and 19917 negative samples, the ratio of positive and negative samples is about 4:1, and the test data set is composed of 24108 samples. In this article, we mainly use Accuracy, F1-scores and AUC as evaluation index [1,4].

5.2. Experimental results and analysis

In order to verify the effectiveness and accuracy of MMSE, we compare it with a single model, the experimental results are shown in Table 2. In order to ensure the accuracy and objectivity of the
experimental results, we conducted six experiments on the six models respectively in the training set, and obtained the average value as the final experimental result.

| model      | accuracy | F1-Score | AUC  |
|------------|----------|----------|------|
| AdaBoost   | 0.8766   | 0.9253   | 0.7556 |
| GBDT       | 0.8767   | 0.9250   | 0.7628 |
| XGBoost    | 0.8745   | 0.9237   | 0.7599 |
| LightGBM   | 0.8776   | 0.9257   | 0.7602 |
| CatBoost   | 0.8770   | 0.9252   | 0.7633 |
| MMSE       | 0.8779   | 0.9260   | 0.8355 |

It can be seen from Table 2 that MMSE model has more advantages than single ensemble learning model in various evaluation indexes. In particular, the improvement in AUC index is relatively large, because MMSE model not only reduces the risk of model overfitting by fusing five ensemble learning models, but also effectively avoids the problem that poor prediction results of a single model lead to excessive deviation of overall prediction of the model.

6. Conclusion and future work
This article focuses on the data analysis and model construction in the dropout prediction problem. Data preprocessing techniques are used to extract 186-dimensional features from the student's log records, and a MMSE dropout prediction model that integrates multiple integrated learning algorithms is constructed. Experimental results show that the model effectively improves the accuracy of student dropout prediction. In the future, we will dig deeper into the factors that affect student dropouts, further enrich the dimensions of existing feature engineering, and hope to improve the performance of the proposed MMSE dropout prediction model and enhance its practicality.

Acknowledgments
This work is supported by National Natural Science Foundation of China (No. U1811264).

References
[1] Taylor C, Veeramachaneni K, O'Reilly U M. (2014) Likely to stop? predicting stopout in massive open online courses[J]. arXiv preprint arXiv:1408.3382.
[2] Liang J, Yang J, Wu Y, et al. (2016) Big data application in education: dropout prediction in edx MOOCs. In: Multimedia Big Data. Taipei. pp. 440-443.
[3] Lu X, Wang S, Huang J, et al. (2017) What Decides the Dropout in MOOCs?. In: Database Systems for Advanced Applications. Chiang Mai. pp. 316-327.
[4] Hong B, Wei Z, Yang Y. (2017) Discovering learning behavior patterns to predict dropout in MOOC. In: Computer Science and Education. Houston. pp. 700-704.
[5] Wang W, Yu H, Miao C. (2017) Deep model for dropout prediction in MOOCs. In: Crowd Science and Engineering. Beijing. pp. 26-32.
[6] P. Fournier-Viger. (2016) The KDDCup 2015 dataset. http://data-mining.philippe-fournier-viger.com/the-kddcup-2015-dataset-download-link/s.