A Dropout Prediction Framework Combined with Ensemble Feature Selection

Dan Ai
School of Computer Science And Engineering
Northeastern University
ShenYang, China
110819
1801767@mail.neu.edu.cn

Tiancheng Zhang*
School of Computer Science And Engineering
Northeastern University
ShenYang, China
110819
tczhang@mail.neu.edu.cn

Ge Yu
School of Computer Science And Engineering
Northeastern University
ShenYang, China
110819
yuge@mail.neu.edu.cn

Xinying Shao
School of Computer Science And Engineering
Northeastern University
ShenYang, China
110819
shaoxinying_neu@163.com

ABSTRACT
In recent years, with the rapid development of large-scale open online courses, low completion rate and high dropout rate have been important challenges for open online courses. Therefore, it is necessary to make effective prediction and timely intervention to ensure the completion of the course. Some of the traditional prediction models only use the features extracted manually from students' clickstream data, which is too subjective to guarantee the quality of features and affect the prediction accuracy. Others generate features automatically with finer granularity, but the problem of feature redundancy appears. In order to solve this problem, this paper proposes a comprehensive dropout prediction framework of MOOCs students. The framework can automatically extract features from clickstream data, and filter features with an integrated feature selection strategy based on clustering and weighted MaxDiff, and finally predict. Experiments show that the model can effectively improve the accuracy of prediction of dropout.

CCS Concepts
• **Applied computing →E-learning**

Keywords
Dropout prediction; ensemble feature selection; Mean-shift clustering; MOOCs.

1. INTRODUCTION
In a few short years, the MOOC movement is in full swing. MOOC arouses more and more learners' enthusiasm for participation with the characteristics of "large-scale", "openness" and "network", which to some extent provides an opportunity for the further development of educational informationization, intelligence education and innovative learning. However, everything has its two sides, MOOC has many advantages in teaching, at the same time, there are also many disadvantages or deficiencies: the number of online learners is large, teachers are difficult to take into account all the students in the classroom, communication and question-answering will be delayed; Distance teaching and evaluation make the teaching quality difficult to guarantee and cheating in exams easier. Although there is a complete teaching pattern, it is still unable to replicate the cultural infiltration and edification given to students by traditional campus culture. There are also deficiencies in networking, community activities and social experience... Among them, the high dropout rate caused by flexible and non-compulsory teaching has become a particularly gloomy phenomenon in the development of MOOC.

In order to make MOOC develop better and bring more benefits in the field of education, how to reduce the high dropout rate has become the focus of MOOC platform, MOOCs teachers and MOOCs investigators [1]. At present, the main methods of MOOCs dropout prediction are based on data mining, that is, feature extraction is carried out on various types of data generated by students in the process of online learning. Finally, multiple mining algorithms are used to predict the dropout of MOOC. Among the various data generated by students in MOOC online learning, clickstream data is the most widely used because it has the widest coverage [2]-[5].

Basically, the user will generate clickstream data more or less during the MOOC online learning process. The clickstream operation data can reflect more details of the learner's learning behavior. We can mine clickstream data to obtain the behavior patterns of learners. The behavior patterns of learners in the current curriculum can better predict whether learners will drop out in the future. In our survey, data mining or feature extraction emerged as a useful part of the integrated model, and especially attracted the attention of MOOCs predictive modeling researchers.

At present, some works have noticed that feature extraction is one of the most difficult and critical tasks in the student's dropout prediction model [6] [7] [8]. Therefore, we can find that effective classification models should pay attention to optimizing feature extraction techniques. Most of the prediction models manually extract a large number of features from clickstream data and then input them to the corresponding classifiers for classification. The features extracted by hand are generally coarse-grained statistics,

***Corresponding author: tczhang@mail.neu.edu.cn**

ICIET 2020, March 28–30, 2020, Okayama, Japan

ACM ISBN 978-1-4503-7705-8/20/03…$15.00
DOI: https://doi.org/10.1145/3395245.3396432

such as [9] [10] and so on, which are calculated on a weekly basis. The effect of the features extracted by these methods depends on the domain knowledge of the extractor. Usually the number of extracted features is small, and the details of the data are not fully utilized, which leads to a certain loss of prediction effect. The number of features extracted from clickstream data in fine granularity (days) is too large, which will cause redundancy of features or the correlation between extracted features and classification is small [11]. In order to improve prediction accuracy and reduce computational complexity, it is necessary to use feature selection method to eliminate useless features. However, feature selection techniques with similar evaluation metrics tend to produce similar feature rankings. If multiple feature selection techniques are combined and several of them are similar, they will dominate the aggregate ranking results, causing the final ranking to be heavily biased toward its choice.

In order to solve these problems, this paper proposes an integrated framework of dropout prediction, which includes feature generation module and an ensemble feature selection module based on clustering and weighted Maxdiff methods.

2. RELATED WORK

2.1 Dropout Prediction

Many scholars at home and abroad have been studying when Students on MOOC platforms will leave the course and dropout. Amnueypornsakul et al. [12] used the clickstream data of learners to predict whether students would dropout. Sort each learner's weekly learning behavior. At the same time, learners are divided into three types: active, abandoned (ie, no learning behavior), and inactive (learning activity sequence is less than two elements). When building a model based on SVM, two types were also constructed: excluding inactive users for forecasting and including inactive users for forecasting. So far, a total of 6 models have been constructed. Each model predicts whether students will drop out next week. Sinha et al. [13] used video click and BBS data to construct the activity sequence of learners and find the footprint sequence that can represent students' active or passive participation in the course. First, construct a learner's weekly learning performance sequence, extract n-gram sequences, video viewing activity sequences and BBS interaction sequences to explore which sequence predicts the learner's dropout behavior more effectively, and which sequence is more representative of the learner's Learning passion. Two experiments were conducted: how this week's learning behavior affects whether stuednts will dropout next week; how the cumulative learning behavior of the first week of the course affects next week's dropout. In addition, some relevant studies are also instructive. For example, Sharkey et al.[14] described in detail the use of machine learning technology to predict dropout iteration process, and obtained predictive characteristics and their relative weight through research. Through the above research, it is found that the clickstream data analysis of students is the main research direction of predict whether students dropout, and the dropout prediction based on the clickstream data extraction is the key research direction of this paper.

2.2 Feature Selection

In real life, an object often has many attributes (called features in this article), which can be roughly divided into three main types: (1) Relevant features: It is helpful for learning tasks (such as classification problems) and can improve the effectiveness of learning algorithms.(2) Unrelated features: Not helpful for learning tasks, and no improvement in algorithm performance. (3) Redundant features: New information is not brought into learning tasks, and information about this feature can be inferred from other functions. Feature selection in the stage of data preprocessing, the validity of preparing data (especially high-dimensional data) for various data mining and machine learning problems has been proved. From the perspective of selection strategy, feature selection methods can be roughly divided into three categories: wrapper methods, filtering methods and embedding methods.

A feature selection method that relies on the predictive performance of a predefined learning algorithm to evaluate the quality of a selected feature is called a wrapper method, such as sequential search [15], mountain climbing search, best priority search [21][16]. The importance of relying on the characteristics of the data to evaluate features is called filtering methods, and their choice is independent of subsequent prediction models. Filter methods are generally more efficient than wrapper methods. However, due to the lack of a special learning algorithm to guide the filtering method in the feature selection stage, the selected features may not be optimal for the target learning algorithm. Related basic algorithms include: chi-square test [18], Pearson correlation coefficient [17] , mutual information [19] , maximum information coefficient (MIC)[20], distance correlation coefficient [17] and so on. The embedding method [17] weighs the filtering method and the wrapper method, which combines feature selection techniques into subsequent prediction models. Therefore, they inherit the wrapper and filtering methods. This method can obtain better feature selection results and prediction accuracy while reducing the time complexity.

3. Our framework

This paper proposes a comprehensive prediction framework of MOOCs students' dropout named FCV-DP, which is mainly composed of the following three parts:

- (1) Feature generation. FCV-DP automatically generates user behavior features in a small time window, which reduces the information loss of the original data in data processing.

- (2) Ensemble feature selection. FCV-DP adopts heterogeneous integration based on clustering and weighted MaxDiff to optimize the limitations of single feature selection techniques.

- (3) Dropout prediction. Various predictive learning devices are used to predict whether students will drop out or not based on the results of integrated feature selection.

This section mainly introduces the specific details of the FCV-DP model.

3.1 Feature generation

The original data set records the type of operation and a specific point in time for each operation by the user. Based on the unit of days, we can automatically extract features from different basic operations of statistical courses, effectively reducing the complexity of manual annotation, and also reducing the information loss of original data in the process of data processing.(See figure 1)

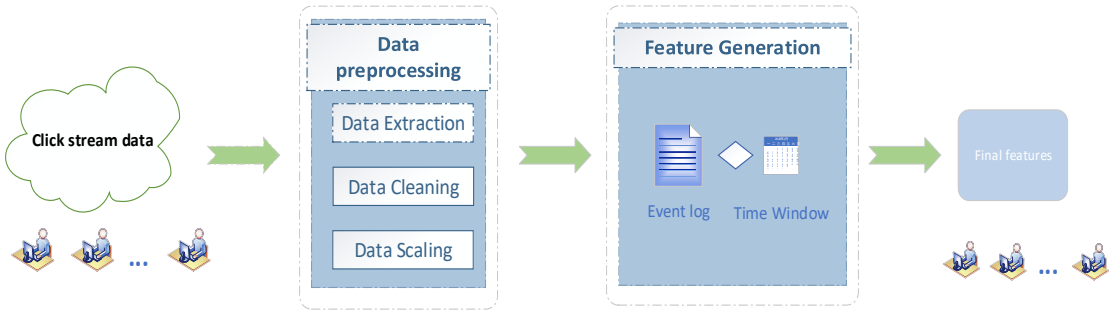

Figure 1. Feature Generation

3.2 Ensemble Feature Selection

A large number of features generated in the feature generation process based on clickstream data may have redundant or classification independent features. In order to improve the accuracy of prediction and reduce the computational complexity, feature selection method is used to eliminate those useless features.

We let F = {F_1, F_2, ..., F_N} represents a candidate feature set with N features. The goal of feature selection model is to select the optimal feature subset S with higher prediction accuracy from the current candidate feature set.

Feature selection methods include filtering type, wrapping type and embedding type. In the filtering type, we use mutual information, chi-square check, T-test and ReliefF.

Mutual information is used for feature selection in classification problems. The more relevant the feature is to the category, the greater the mutual information value. Suppose the joint distribution of features and categories is p(x,y), the marginal distribution of features and categories is p(x) and p(y), the mutual information I(x;y) is the relative entropy of joint distribution p(x) and the marginal distribution p(y).

$$I(X:Y)=\sum_{x\in X}\sum_{y\in Y}p(x,y)\log_2\frac{p(x,y)}{p(x)p(y)}$$

In the filtering method, it is generally assumed that the features independent of labels are irrelevant features, while chi-square test can be used for independent test, so it is suitable for feature selection. Chi-square test is the deviation degree between the actual observed value and the theoretical inferred value of the statistical sample.

T -score [27] is used for binary classification problems. For each feature f_i, suppose that μ_1 and μ_2 are the mean feature values for the instances from two different classes, σ_1 and σ_2 are the corresponding standard deviations, n_1 and n_2 denote the number of instances from these two classes. Then the t_score for the feature f_i is:

$$t\_score(f_i)=|\mu_1-\mu_2|/\sqrt{\frac{\sigma_1^2}{n_1}+\frac{\sigma_2^2}{n_2}}$$

The basic idea of T-score is to assess whether the feature makes the means of two classes statistically different, which can be computed as the ratio between the mean difference and the variance of two classes. The higher the t_score, the more important the feature is.

ReliefF algorithm was first proposed by Kira [26]. The basic content of ReliefF algorithm is to randomly select a sample R from the training set D, then find the k-nearest neighbor samples H from the samples of the same category as R, find the k-nearest neighbor sample M from the samples of different categories from R, and finally update the feature weight according to the formula.

The weight of each feature f_i in ReliefF is defined as follows:

$$w(f_i)=w(f_i)-\sum_{j=1}^{k}diff(f_i,R,H_j)/k+\sum_{C\in class(R)}\left[\frac{p(C)}{1-p(Class(R))}\sum_{j=1}^{k}diff(A,R,M_j(C))\right]/k$$

Where $M_j(C)$ represents the k-nearest neighbor sample in class C, diff(A,R_1,R_2) represents the difference between samples on feature A.

$$diff(A,R_1,R_2)=\begin{cases}\dfrac{|R_1[A]-R_2[A]|}{\max(A)-\min(A)} & if\ \ A\ \ is\ \ continuous\\ 0 & if\ \ A\ \ is\ \ discrete\ \ R_1[A]=R_2[A]\\ 1 & if\ \ A\ \ is\ \ discrete\ \ R_1[A]\neq R_2[A]\end{cases}$$

The filtering method uses statistical indicators to score and screen each feature, focusing on the characteristics of the data itself. The advantage is that the calculation is fast and does not depend on specific models. The disadvantage is that the selected statistical indicators are not customized for specific models, so the final accuracy may not be high. Because it is a univariate statistical test, the correlation between features is not considered.

In contrast, wrapper methods take feature selection as a feature subset search problem, screen various feature subsets, and evaluate the effect with model. Recursive feature elimination method uses a basic model to carry out multiple rounds of training. After each round of training, the features of several weight coefficients are eliminated, and then the next round of training is carried out based on the new feature set.

However, the embedded type feature selection algorithm uses the base model with regularization, which not only screens out the feature, but also reduces the dimension. Regularization is to add additional constraints or penalties to the existing model (loss function) to prevent over-fitting merging and improve generalization ability.

FCV-DP model uses the basic feature selection models to generate feature rankings, and then integrates them. Because feature selection techniques with similar standards tend to produce similar output [28][29]. If multiple FS techniques are combined simply and several are similar, they will dominate in aggregation

and the resulting output will be strongly biased toward their choice. This outcome can be avoided by careful selection of the FS methods for an ensemble or robust voting approach. However, which FS methods have similar backgrounds may not be apparent, and it is difficult to design a robust voting approach that works in every scenario. Therefore, in addition to a conventional heterogeneous ensemble, we also propose a clustered ensemble. After sorting the output using the basic FS method, the output is clustered to identify similar FS outputs (or more similar outputs) and group them. Since similar FS outputs are clustered together, they have less chance to over-vote the other methods, which enhances diversity. There are many kinds of clustering methods, this paper uses the Mean-Shift clustering algorithm.

Most of the commonly used clustering algorithms such as K-means algorithm, K-medoids algorithm, have to determine the number of clusters in advance, the commonly used elbow rule is not omnipotent, always encounter difficult choice. However, one of the advantages of Mean-Shift clustering algorithm is that it can automatically determine the number of clusters according to the sample density, without the need to pre-specify the number of clusters artificially. Mean-Shift algorithm was first proposed by Fukunage in 1975, and then extended by Yizong Cheng, Two main improvements are proposed:(1) The kernel function is defined. The definition of the kernel function makes the contribution of the offset value to the offset vector vary with the distance between the sample and the offset point. The definition of Gaussian kernel function used in this paper is as follows, where h is called bandwidth.

$$N(x)=\frac{1}{\sqrt{2\pi}h}e^{-}\frac{x^2}{2h^2}$$

Afte$_r$ introducing the kernel function into the mean shift vector, the resulting mean shift vector form is as follows:

$$M_h(x)=\frac{\sum_{i=1}^{n}G(\frac{x_i-x}{h_i})(x_i-x)}{\sum_{i=1}^{n}G(\frac{x_i-x}{h_i})}$$

(2) The weight coefficient is increased. Considering the influence of the distance of different samples on mean shift vector, we can introduce a weight $\omega(x_i)$ for each sample point x_i, satisfy $\sum\omega(x_i)=1$, $\omega(x_i)>0$, The vector form can be extended as follows:

$$M_h(x)=\frac{\sum_{i=1}^{n}G(\frac{x_i-x}{h_i})\omega(x_i)(x_i-x)}{\sum_{i=1}^{n}G(\frac{x_i-x}{h_i})\omega(x_i)}$$

The weight coefficients make the weights of different samples different. Mean-Shift algorithm is widely used in clustering, image smoothing, segmentation and video tracking.

By means of Mean-Shift algorithm, we can get several different feature rankings. The next step is to merge these rankings into a single one. Methods designed to combine several ranked lists into a single final decision are, in general, known as ensemble or rank aggregation techniques. The ranking synthesis problem can be described by a voting model, in which the ranking results obtained by Mean-Shift clustering are regarded as voters and all the features are considered as candidates. The basic idea of the graph theory algorithm based on the voting model and the corresponding ranking synthesis is to give a candidate (feature) set F={f_1, f_2, ..., f_n}. A sort r of F is a rank of all the candidates in F, representing the preference of a voter for a candidate. For $f_i \in F$, $r_k(f_i)$ represents the position of feature f_i in sort v_k. For any $f_{i,}$ $f_j \in F$, $r_k(f_i)<r_k(f_j)$ indicates that f_i ranks higher in rank v_k than f_j. There are n sorts $v_1,v_2...v_n \in K$ representing the sort given by n voters, and the sort synthesis algorithm based on the voting model is to compute the synthetic sort M of the k sorts for n candidates.

Voting algorithm can be described by graph theory. A digraph G={N,E} is provided, where in the vertex set N corresponds to a candidate set, an arc (x,y)∈E between any two vertices.

$$\omega(x,y)=\sum_{v_i\in k}\left|r_i(y)>r_i(x)\right|$$

$$In\deg ree(x)=\sum_{m\in N,m\neq x}\omega(m,x)$$

$$Out\deg ree(x)=\sum_{m\in N,m\neq x}`\omega(x,m)$$

ω(x,y) is the weight of the edge，Indegree(x) and Outdegree(x) represent the in-degree and out-degree of the vertex.

The basic idea of elimination voting is to eliminate the eliminated candidates in turn until the winner is elected. The common way to eliminate votes is to eliminate the “best” candidate in turn or the “worst” one in turn. Our algorithm Maxdiff considers the degree of vertex in and out. In each iteration of the algorithm, the absolute value of the difference between the in-degree and the out-degree of the vertex is calculated as the judging condition, and the vertex with the largest absolute value is eliminated.

$$b_{F_i}=\text{Outdegree}\left(F_i\right)-\text{Indegree}\left(F_i\right)$$

The candidate with the largest difference between the two is the best or worst candidate at present.

Because the selection of linear scoring system is arbitrary, it is not suitable for feature selection. Therefore, it is not very accurate to convert sequential ordering directly into linear fractional preference. We used the weighted modification of MaxDiff.

In this paper, the discrete step function is defined as the following formula.

$$u[n]=\begin{cases}1, n>0\\0, n<0\end{cases}$$

Where $n\in N$, two parameters need to be set here: the weighted characteristic number M and the step number Q. if the step size is L=M/Q, the weighted vector β is defined as follows:

$$\beta[n]=\sum_{q=1}^{Q}u[q\bullet L-n]$$

This paper defines the operations arranged in descending order of vector elements as functions g(.), its reverse operation returns the original sorting of elements. Therefore, the final weight calculation formula is as follows:

$$S_k^{'} = g(v_k)$$

$$\omega'(x,y) = \sum_{v_k \in K}((g^{-1}(S_k^{'} \otimes \beta))^y - (g^{-1}(S_k^{'} \otimes \beta))^x)$$

Where $(.)^x$ represents the corresponding value of vertex x. See Figure 2 for the overall architecture of ensemble feature selection.

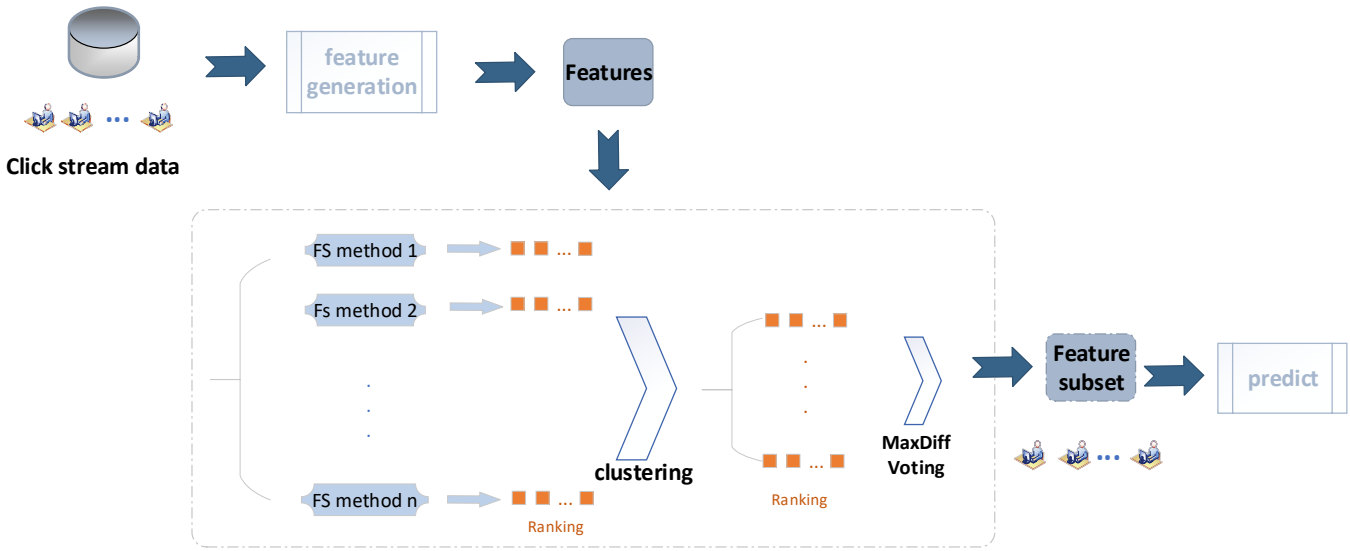

Figure 2. General framework

3.3 Prediction

When the classifier selected by the recursive feature elimination algorithm is consistent with the final predictive classifier, the accuracy of the final predictive result can reach the maximum. For ease of explanation, we choose the same classifier. For each test case, after selecting the optimal feature subset, the test data is transformed to extract features, which is mapped to the corresponding space of the optimal feature subset, and then the pre-trained model is used to predict the corresponding categories.

4. Experiments and results

In this section, we carry out a detailed experiment based on a real MOOC data set to prove the effectiveness of our proposed framework, and give the experimental results and analysis. This article uses the clickstream data set of the MOOC platform provided by 2015 KDD CUP. If a student does not operate on a course of his choice for ten consecutive days , the data set marks the user-course item as dropping out. The data set includes 39 courses, 79,186 students, and 120,542 enrolled in 39 courses. There are 8,157,277 entries in the 30-day activity log for each course.(see Table 1)

Table 1. Data set element item

Attribute	Description
Enrollment_id	The enrollment record Identification, that is associated with which student enrolls in which course
Event	Event type
Object	The object that the student access

The data set defines seven different event types, its exact name and meaning are shown in Table 2

Table 2. Event Type

Type	Description
Problem	Do homework
Video	Watch the video
Access	Access other objects except the video and the job
Wiki	Read the Wikipedia of the course
Discussion	Forum discussion
Navigate	Navigate other objects except the video and the job
Page close	Close the web page

In the feature generation part of the model, the model automatically generates 210 features in the time window of day. Then we feed the generated features into the integrated feature selection framework proposed in this paper and compare it with the baseline feature selection methods. In order to make the experimental results more referential, we use three common machine learning classification method: AdaBoost, RandomForest, NaiveBayes in prediction model.

Table 3, table 4 and table 5 show the best performance of the model under different evaluation indexes when the number of selected features does not exceed 50. It can be seen that the dropout prediction accuracy using integrated feature selection in this model is better than embedding other baseline feature selection methods in this model. Practice has proved that the strategy proposed in the feature selection part of the FCV-DP model is effective. The strategy uses heterogeneous integration methods based on clustering and weighted voting to optimize the limitations of single feature selection techniques. The table proves that the model proposed in this paper can reduce the number of

features and save the calculation time while achieving a good prediction accuracy.

Table 3. Performance comparison between FCV-DP and baseline method using SVM prediction

Model	Auc	F1	Recall
ANOVA	**0.83**	**0.90**	**0.99**
CHI	**0.78**	**0.87**	**0.95**
MI	**0.78**	**0.87**	**0.95**
MyRFE	**0.81**	**0.88**	**0.97**
Relief	**0.81**	**0.87**	**0.81**
Ttest	**0.79**	**0.88**	**0.98**
FSPred	**0.79**	**0.87**	**0.91**
FCV-DP without clustering	**0.78**	**0.87**	**0.95**
FCV-DP	**0.84**	**0.90**	**0.99**

Table 4. Performance comparison between FCV-DP and baseline method using AdaBoost prediction

Model	Auc	F1	Recall
ANOVA	0.82	0.90	0.99
CHI	0.79	0.87	0.92
MI	0.79	0.87	0.92
MyRFE	0.83	0.89	0.97
Relief	0.84	0.90	0.98
Ttest	0.79	0.87	0.94
FSPred	0.79	0.87	0.91
FCV-DP without clustering	0.83	0.88	0.92
FCV-DP	0.84	0.90	0.99

Table 5. Performance comparison between FCV-DP and baseline method using NaiveBayes prediction

Model	Auc	F1	Recall
ANOVA	0.82	0.89	0.99
CHI	0.78	0.86	0.90
MI	0.78	0.86	0.90
MyRFE	0.81	0.88	0.97
Relief	0.81	0.87	0.84
Ttest	0.78	0.86	0.92
FSPred	0.79	0.88	0.95
FCV-DP without clustering	0.78	0.86	0.90
FCV-DP	0.84	0.90	0.99

5. CONCLUSION

In this paper, an integrated framework of dropout prediction is proposed, which includes feature selection module based on clustering and weighted MaxDiff. The framework has been effective in tackling the problem of predicting dropouts from large-scale open online courses (MOOCs). The model uses the integrated feature selection method based on clustering and weighted MaxDiff to find the optimal feature subset. Finally, classifiers are used to predict dropouts. Experiments show that the model can extract effective features from the MOOCs students' click-stream data set and find the best feature subsets, which improves the prediction accuracy and reduces the computational complexity.

6. ACKNOWLEDGMENTS

This work is supported by National Nature Science Foundation under Grant (Nos.U1811261,61602103) and the Fundamental Research Funds for the Central Universities (N180716010).

7. REFERENCES

[1] Y.Wang, "Exploring possible reasons behind low student retention rates of massive online open courses: A comparative case study from a social cognitive perspective," in Proc.1st Workshop Massive Open Online Courses,16th Annu. Conf. Artif. Intell. Educ., Jun. 2013, p. 58.

[2] C. Taylor, K. Veeramachaneni, and U.-M. O'Reilly. (2014). Likely to stop?Predicting stopout in massive open online courses. [Online]. Available:https://arxiv.org/abs/1408.3382

[3] D. S. Chaplot, E. Rhim, and J. Kim, "Predicting student attrition in MOOCs using sentiment analysis and neural networks," in Proc. Workshop Intell. Support Learn. Groups, 2015, pp. 7-12.

[4] M. Fei and D.-Y. Yeung, "Temporal models for predicting student dropout in massive open online courses," in Proc. IEEE Int. Conf. Data Mining Workshop (ICDMW), Nov. 2015, pp. 256-263.

[5] J.-L. Hung, M. C. Wang, S. Wang, M. Abdelrasoul, Y. Li, and W. He, "Identifying at-risk students for early interventions-A time-series clustering approach," IEEE Trans. Emerg. Topics Compute., vol. 5, no. 1,pp. 45-55, Jan./Mar. 2017.

[6] Li,W., Gao, M., Li, H., Xiong, Q.,Wen, J.,Wu, Z.: Dropout prediction in MOOCs using behavior features and multi-view semi-supervised learning. In: 2016 International Joint Conference on Neural Networks (IJCNN), pp. 3130–3137 (2016a)

[7] Robinson, C., Yeomans, M., Reich, J., Hulleman, C., Gehlbach, H.: Forecasting student achievement in MOOCs with natural language processing. In: Proceedings of the Sixth International Conference on Learning Analytics and Knowledge, LAK '16, pp. 383–387. ACM, New York (2016)

[8] Nagrecha, S., Dillon, J.Z., Chawla, N.V.: MOOC dropout prediction: lessons learned from making pipelines interpretable. In: Proceedings of the 26th International Conference on World Wide Web Companion,International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva,Switzerland, WWW '17 Companion, pp. 351–359 (2017)

[9] G. K. Balakrishnan and D. Coetzee, ``Predicting student retention in massive open online courses using hidden Markov models," Dept. Elect. Eng. Comput. Sci., Univ. California, Berkeley, CA, USA, Tech.Rep. UCB/EECS-2013-109, May 2013.

[10] S. Halawa, D. Greene, and J. Mitchell, ``Dropout prediction in MOOCs using learner activity features," in Proc. MOOCs, vol. 7, 2014, pp. 3-12.

[11] Qiu L, Liu Y, Liu Y. An integrated framework with feature selection for dropout prediction in massive open online courses[J]. IEEE Access, 2018, 6: 71474-71484.

[12] Amnueypornsakul B, Bhat S, Chinprutthiwong P. Predicting Attrition Along the Way: The UIUC Model [C]// Proceedings Of the EMNLP 2014 Workshop on Analysis of Large Scale Social Interaction in MOOCs, Doha, Qatar. Association for Computational Linguistics, 2014: 55-59.

[13] Sinha T, Jermann P, Li N, et al. Your Click Decides Your Fate: Inferring Information Processing and Attrition Behavior from MOOC Video Clickstream Interactions[OL]. arXiv Preprint. arXiv:1407.7131, 2014.

[14] Sharkey M, Sanders R. A Process for Predicting MOOC Attrition[C]//Proceedings of the EMNLP 2014 Workshop on Analysis of Large Scale Social Interaction in MOOCs,Doha, Qatar. Association for Computational Linguistics, 2014: 50-54.

[15] Guyon I . An introduction to variable and feature selection[M]. JMLR.org, 2003.20

[16] Arai H , Maung C , Xu K , et al. Unsupervised feature selection by heuristic search with provable bounds on suboptimality[C]// Thirtieth Aaai Conference on Artificial Intelligence. AAAI Press, 2016.

[17] Guyon I , Elisseeff, Andr é An Introduction to Variable and Feature Selection[J]. Journal of Machine Learning Research, 2003, 3(6):1157-1182.

[18] And H L , Liu H , Setiono R . Chi2: Feature Selection and Discretization of Numeric Attributes[C]// International Conference on Tools with Artificial Intelligence. IEEE Computer Society, 1995.

[19] Hoque N , Bhattacharyya D K , Kalita J K . MIFS-ND: A mutual information-based feature selection method[J]. Expert Systems with Applications, 2014, 41(14):6371-6385.

[20] Lei Y, Liu H. Feature selection for high-dimensional data: a fast correlation-based filter solution[C]// Twentieth International Conference on International Conference on Machine Learning. 2003.

[21] Kohavi R , John G H . Wrappers for Feature Subset Selection[J]. Artificial Intelligence, 1997, 97(1-2):273-324.

[22] Kira K, Rendel L. The feature selection problem: traditional methods and a new algorithm[J]. Proc. AAAI-92, 1992.

[23] Davis J G. STATISTICS AND DATA ANALYSIS IN GEOLOGY[M]// Statistics and data analysis in geology. 1973.

[24] Dess ìN, Pes B. Similarity of feature selection methods: An empirical study across data intensive classification tasks[J]. Expert Systems with Applications, 2015, 42(10):4632-4642.

[25] Drot ár, P, Gazda J , Sm ékal, Z. An Experimental Comparison of Feature Selection Methods on Two-Class Biomedical Datasets[J]. Computers in Biology and Medicine, 2015, 66:S0010482515002917.