Front. Comput. Sci., 2020, 14(5): 145313
https://doi.org/10.1007/s11704-019-9062-8

RESEARCH ARTICLE

Multi-task MIML learning for pre-course student performance prediction

Yuling MA[1,2], Chaoran CUI (✉)[3], Jun YU[1], Jie GUO[1], Gongping YANG[1], Yilong YIN (✉)[1]

1 School of Software, Shandong University, Jinan 250100, China
2 School of Information Engineering, Shandong Yingcai College, Jinan 250104, China
3 School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan 250014, China

Abstract In higher education, the initial studying period of each course plays a crucial role for students, and seriously influences the subsequent learning activities. However, given the large size of a course's students at universities, it has become impossible for teachers to keep track of the performance of individual students. In this circumstance, an academic early warning system is desirable, which automatically detects students with difficulties in learning (i.e., at-risk students) prior to a course starting. However, previous studies are not well suited to this purpose for two reasons: 1) they have mainly concentrated on e-learning platforms, e.g., massive open online courses (MOOCs), and relied on the data about students' online activities, which is hardly accessed in traditional teaching scenarios; and 2) they have only made performance prediction when a course is in progress or even close to the end. In this paper, for traditional classroom-teaching scenarios, we investigate the task of pre-course student performance prediction, which refers to detecting at-risk students for each course before its commencement. To better represent a student sample and utilize the correlations among courses, we cast the problem as a multi-instance multi-label (MIML) problem. Besides, given the problem of data scarcity, we propose a novel multi-task learning method, i.e., MIML-Circle, to predict the performance of students from different specialties in a unified framework. Extensive experiments are conducted on five real-world datasets, and the results demonstrate the superiority of our approach over the state-of-the-art methods.

Keywords educational data mining, academic early warning system, student performance prediction, multi-instance multi-label learning, multi-task learning

1 Introduction

For college students, one of the most basic and important tasks is studying courses. It is widely accepted that the initial period of learning a new course is crucial for students [1, 2]. During this period, students can experience the novelty of the course, eliminate doubts, and lay the foundation for the follow-up learning stages. However, owing to certain difficulties (e.g., course materials are difficult to understand), some students may lose interest or even give up on studying at this stage, which seriously influences the subsequent learning activities. Moreover, given the large size of a course's students at universities, it has become impossible for teachers to keep track of the performance of individual students. In this circumstance, an academic early warning system is desirable, which can automatically detect at-risk students (i.e., students who may have difficulty with a certain course) prior to a course's commencement.

As the key issue in developing academic early warning systems, student performance prediction aims to estimate students' performance from various aspects, such as scores, ranks and grades, which can be either numerical/continuous

Received February 19, 2019; accepted July 23, 2019

E-mail: crcui@sdufe.edu.cn; ylyin@sdu.edu.cn

value (regression task) or categorical/discrete value (classification task) [3]. However, despite considerable research on student performance prediction, existing methods have two major limitations. Firstly, many studies are concerned with e-learning platforms, including massive open online courses (MOOCs) [4, 5], intelligent tutoring systems (ITS) [6], learning management systems (LMSs) [7–10], and hellenic open university (HOU) [11, 12]. They heavily rely on the online activities of students, which might not be available in traditional classroom-teaching scenarios [13]. Secondly, most existing methods can only make predictions when a course is in progress [13, 14] or even close to the end [15, 16]. They are ineffective in helping students in the early learning period.

In this paper, we focus on the traditional classroom-teaching scenes, and seek to predict students' performance prior to the start of each course. Therefore, we term our research *pre-course student performance prediction*. Intuitively, a student's performance on previous courses is highly related to that on new courses. For example, if a student has achieved an excellent performance on the course "operating system", it is much likely that he/she will perform well on the course "distributed operating system" as well [17]. Motivated by this, we propose to leverage students' performance in past semesters to predict their performance on future courses. However, this idea faces three main challenges:

- Owing to the existence of optional courses, the records of completed courses may be inconsistent across students. As a result, students cannot be simply represented in a common feature space.
- There are multiple courses offered in a new semester, which are generally correlated with each other. Therefore, instead of predicting students' performance on each course separately, all the target courses should be considered as a whole.
- There lacks large-scale public datasets for pre-course student performance prediction. This impedes the development of learning-based prediction methods, which generally have certain requirements on the sample size to achieve good prediction performance [18].

To address the above issues, we cast the task of pre-course student performance prediction as a multi-instance multi-label (MIML) problem [19]. Specifically, in multi-instance representation, we treat each student as a bag of instances, each of which represents the information of a specific previous course of the student. In this way, the problem of inconsistent course histories across students is fully resolved. In multi-label prediction, we treat target courses as labels and predict them simultaneously. In this way, the correlations between courses are implicitly utilized. Besides, we collect a new group of MIML datasets for pre-course student performance prediction. Given the limited amount of samples in each dataset, we propose a multi-task learning method, namely MIML-Circle. Multi-task learning aims to leverage useful information contained in multiple related tasks to help improve the generalization performance of all the tasks via learning them jointly. It is an empirically good solution, particularly when training samples of each related task are considerably limited [20]. In MIML-Circle, multiple models can be jointly learned on different MIML datasets. In order to exploit the benefits from other related tasks, the labels of a sample predicted by all classifiers (i.e., including classifiers of a task itself and those of other tasks) are utilized as new features of the sample. Then it builds predictive models iteratively with these augmented features.

Our main contributions can be summarized as three-fold:

(1) We investigate the problem of academic early warning from a new perspective of pre-course student performance prediction.
(2) We cast the task as an MIML learning problem to make full use of the historical course information of students, as well as the correlations among multiple target courses.
(3) We collect a new group of datasets for pre-course student performance prediction, and propose a novel multi-task learning method to alleviate the data scarcity problem.

The remainder of this paper is organized as following. Section 2 reviews the related work. Section 3 details our framework for pre-course student performance prediction. Experimental results and analysis are reported in Section 4, followed by the conclusion and future work in Section 5.

2 Related work

As one of the most important and popular topics in educational data mining, student performance prediction has drawn numerous research attention in recent decades. Owing to the convenience of collecting data, the majority of existing research has been concerned with e-learning platforms, including MOOCs [4, 5], ITS [6], LMSs [7–10], HOU [11, 12], and other platforms [21–24]. For example, Ren et al. predicted grades using data from MOOC server logs, such as the average number of daily study sessions, total video viewing time, number of videos a student watches, and number of quizzes [5]. Macfadyen and Dawson developed predictive models of student final grades, based on LMS tracking data, including the number of discussion messages posted, number of mail messages sent, and number of assessments completed [9]. Zafra et al. predicted students' performance (i.e., pass or fail) with the information about quizzes, assignments and forums stored in Moodle, which is a free learning management system [10]. As can be seen, the above studies for e-learning platforms have mainly relied on the data about students' online activities, which is hardly accessed in traditional classroom-teaching scenes.

For traditional classroom-teaching environments, most studies can only make predictions when target courses are in progress or even close to the end. Marbouti et al. utilized the in-semester performance factors, including grades for attendance, quizzes, and weekly homework, to predict at-risk students after the fifth week of the semester [13]. Meier et al. predicted students' final grades after the forth course week with the performance assessments on homework assignments, midterm exam, course project, and final exam [14]. Some studies [15, 16] could not predict the final grade of a student until half of a semester passed, because they relied on the results of the mid-semester quiz. Therefore, these studies are ineffective in helping students in the early learning period.

The most related work to ours is [1]. In [1], matrix completion methods were conducted to predict grades for each student for the next enrollment term based on grades information that students earned on completed courses. Although this research can predict student performance prior to a course's commencement, it works from the perspective of recommender systems and greatly differs from our study.

3 Framework

In this section, we first illustrate the framework of pre-course student performance prediction with MIML learning. Then, in order to solve the problem of data scarcity, we introduce a novel multi-task learning method , i.e., MIML-Circle, for our task.

To formulate our problem, we use capital letters (e.g., X), bold lowercase letters (e.g., $\mathbf{x}$), and non-bold lowercase letters (e.g., x) to denote sets, vectors, and scalars, respectively. Table 1 summarizes the key notations and definitions used throughout the article.

Table 1 Summary of key notations and definitions

Notation	Definition
i, j, k, t	index variables
$S_i = (X_i, Y_i)$	a student sample
X_i	a bag of instances to describe the student S_i
Y_i	the label set (i.e., difficult courses) of the student S_i
n_i	the number of instances in X_i
c_i	the number of labels in Y_i
$\mathbf{x}_j \in X_i$	a vector describing one of a student's finished courses
$y_k \in Y_i$	a class label corresponding to a difficult course of S_i
D_{MIML}	an MIML student dataset $\{(X_i, Y_i)\|i = 1, 2, \ldots, n\}$
m	the number of tasks
D^t_{MIML}	the MIML dataset $\left\{(X^t_i, Y^t_i)\|i = 1, 2, \ldots, n^t\right\}$ for the tth task
$\phi(.)$	a mapping of transforming multi-instance samples into single-instance samples
D^t_{SIML}	the SIML dataset $\left\{(\phi(X^t_i), Y^t_i)\|i = 1, 2, \ldots, n^t\right\}$ for the tth task
f^t_{MLSVM}	a multi-label classifier constructed on D^t_{SIML}
F	an MLSVM classifier set, i.e., $F = \bigcup_{t=1}^{m} f^t_{MLSVM}$
$\mathbf{l}$	the predictive label vector of a sample predicted by all classifiers in F
L	predictive label vectors set
P^t	a student sample for testing in the tth task
R	the maximum iteration number
θ	$0 \leqslant \theta \leqslant 1$, the decision parameter

3.1 MIML learning

As aforementioned, owing to the existence of optional courses, students cannot be simply represented in a common feature space. In addition, target courses are generally correlated and thus should be considered as a whole. As a result, traditional supervised learning framework may be unsuitable, in which samples need to be represented over a common feature space and different class labels are predicted separately. To address this issue, we cast the task of pre-course student performance prediction as an MIML learning problem.

In MIML learning framework, each student sample in our study is described as $S_i = (X_i, Y_i)$, where X_i is a bag of instances $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n_i}\}$, and $Y_i = \{y_1, y_2, \ldots, y_{c_i}\}$ is the label set of S_i. Specifically, $\mathbf{x}_j \in X_i$ $(j = 1, 2, \ldots, n_i)$ is an instance (i.e., a single vector) describing one of the student's finished courses, e.g., "Periods: 64 (hours), Theory-teaching period: 32 (hours), Experiment period: 32 (hours), Credit: 4, Course nature: 1 (1 compulsory or 0 optional), Examination form: 1 (1 close-book or 0 open-book), and Score: 80". n_i denotes the number of instances in X_i, and for different student samples, the value of n_i can be different. The label $y_k \in Y_i$ $(k = 1, 2, \ldots, c_i)$ represents a difficult course of the student S_i, in which c_i denotes the number of labels in Y_i. Given an

MIML student dataset $D_{MIML} = \{(X_i, Y_i) \mid i = 1, 2, \ldots, n\}$, we can construct predictive models with MIML algorithms. Over the past few years, various MIML algorithms have been developed [25–31]. In this work, we focus on the MIMLSVM algorithm because of its favorable balance between accuracy and efficiency [19]. Note that more complicated MIML algorithms can also be adopted here, but we leave them for future exploration.

MIMLSVM tackles an MIML problem by identifying its equivalence in the traditional supervised learning framework, using multi-label learning as the bridge [19]. It constructs predictive models in two steps: 1) transform the multi-instance to a single-instance representation; and 2) construct SVM models via multi-label learning method. Given $D_{MIML} = \{(X_i, Y_i) \mid i = 1, 2, \ldots, n\}$, in which n is the number of samples, MIMLSVM first transforms it into a single-instance multi-label (SIML) dataset $D_{SIML} = \{(\mathbf{z}_i, Y_i) \mid i = 1, 2, \ldots, n\}$. Here, $\mathbf{z}_i = \phi(X_i)$ is a single vector, in which $\phi(\cdot)$ is a mapping of transforming multi-instance samples into single-instance samples. In MIMLSVM, the *constructive clustering* algorithm is adopted to obtain $\mathbf{z}_i$. The details of *constructive clustering* can be found in [32]. Based on D_{SIML}, a multi-label method called MLSVM [33] is then utilized to construct predictive models, which trains SVM classifiers by decomposing the multi-label learning problem into multiple independent binary classification problems.

3.2 MIML-Circle

Due to the lack of publicly available datasets, our study is carried out on a new group of self-collected datasets, which will be described in detail later. However, given the limited amount of samples in each dataset, it is very challenging to construct accurate and promising predictive models. To address this issue, we introduce multi-task learning [20] into our study. Considering that there are different course settings for different specialties, we regard the construction of predictive models on datasets generated from different specialties as different tasks. A novel multi-task learning method called MIML-Circle is proposed for MIML learning scenes, which constructs predictive models for different tasks simultaneously, and improves the performance of each model via exploiting the relatedness of different tasks.

MIML-Circle is proposed following the stacking idea, in which the core principle is to train classifiers using the original training data set at first, and then the outputs of the classifiers are regarded as input features to train new classifiers. More detailed information about stacking can be found in [34]. Specifically, MIML-Circle first constructs classifiers on each MIML data set, and the predictive labels by these classifiers are utilized as new features to augment the original data sets, and thus classifiers can be retrained with the augmented data [35]. As shown in Fig. 1, the training process of MIML-Circle can be divided into the following three steps:

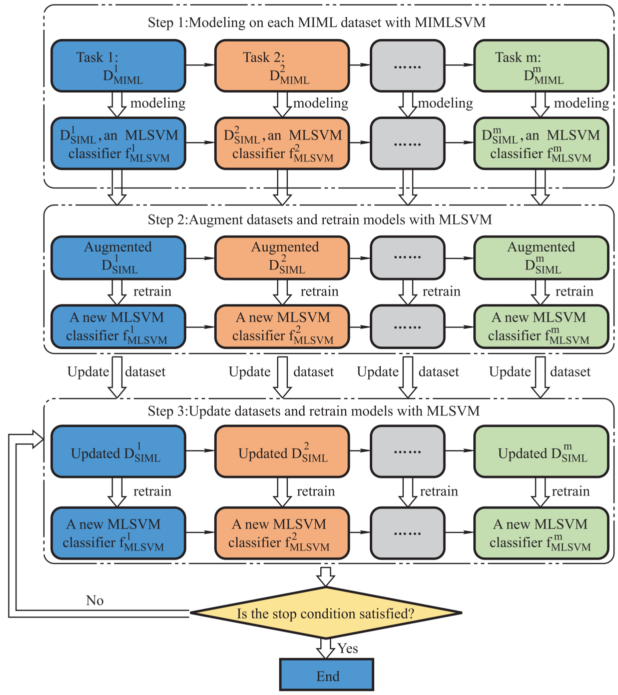

Fig. 1 The flow diagram of MIML-Circle

(1) Model on each MIML dataset with the MIMLSVM method. Given m MIML datasets $D^t_{MIML} = \left\{(X^t_i, Y^t_i) \mid i = 1, 2, \ldots, n^t;\ t = 1, 2, \ldots, m\right\}$ corresponding to m different tasks, where n^t is the number of samples in the tth task, we randomly link all the datasets as a chain, and utilize MIMLSVM to construct the predictive models on each dataset in turn. In this step, for each task t, its MIML dataset is transformed to a SIML dataset $D^t_{SIML} = \left\{(\mathbf{z}^t_i, Y^t_i) \mid i = 1, 2, \ldots, n^t\right\}$, and then an MLSVM classifier f^t_{MLSVM} can be generated.

(2) Augment datasets and retrain models. We first augment the SIML dataset D^t_{SIML} by incorporating the prediction labels of $\mathbf{z}^t_i$ into the representation $\mathbf{z}^t_i$. Specifically, for a SIML sample $(\mathbf{z}, Y)$, we augment its representation $\mathbf{z}$ by linking with the predictive labels from all classifiers $\left\{f^1_{MLSVM}, f^2_{MLSVM}, \ldots, f^m_{MLSVM}\right\}$, which were generated in the previous step (i.e., the initial values for the additional features assigned by other tasks' classifiers). In other words, the new representation $\mathbf{z}'$ can be obtained

as:

$$\mathbf{z}' = [\mathbf{z}, \mathbf{l}], \tag{1}$$

$\mathbf{l} = \boldsymbol{Pre}(F, \mathbf{z})$ denotes the prediction label vector of the input $\mathbf{z}$ outputted by the classifier set $F = \left\{f^1_{MLSVM}, f^2_{MLSVM}, \ldots, f^m_{MLSVM}\right\}$. Based on the augment datasets $D^t_{SIML} = \left\{(\mathbf{z}'^t_i, Y^t_i) | i = 1, 2, \ldots, n^t\right\}$, we retrain models f^t_{MLSVM} with the MLSVM method.

(3) Update datasets and retrain models. Relying on these new models obtained in step (2), for each sample $\mathbf{z}'$, we get its new predictive label vector $\mathbf{l}$, and update the augmented features of $\mathbf{z}'$. Then, we retrain models on these updated datasets. This step is conducted iteratively until the termination condition is satisfied. In this study, we denote R as the maximum iteration number. The pseudo code of MIML-Circle is given in Algorithm 1.

Algorithm 1 The pseudo code of MIML-Circle

```
Input:
    m MIML training sets:
    D^t_MIML = {(X^t_i, Y^t_i) | i = 1, 2, ..., n^t; t = 1, 2, ..., m};
Output:
    the MLSVM classifier sets:
    F_1, F_2, and all F_new generated in the step (3)
(1) MIMLSVM step
for t = 1 to m
    [f^t_MLSVM, D^t_SIML] ← MIMLSVM(D^t_MIML);
end for
F_1 ← {f^1_MLSVM, f^2_MLSVM, ..., f^m_MLSVM};
(2) Augment datasets and retrain models
for t = 1 to m
    Augment D^t_SIML according to Eq. (1);
    [f^t_MLSVM] ← MLSVM(D^t_SIML);
end for
F_2 ← {f^1_MLSVM, f^2_MLSVM, ..., f^m_MLSVM};
(3) Update datasets and retrain models
F_old = F_2; D^t_old = D^t_SIML;
while the termination condition is not satisfied
    for t = 1 to m
        L ← {l | l = Pre(F_old, x), ∀x ∈ D^t_old};
        D^t_new ← Update D^t_old with new label vectors in L;
        f^t_new ← Retrain models on D^t_new with MLSVM;
    end for
    F_new ← {f^1_new, f^2_new, ..., f^m_new};
    F_old ← F_new; D^t_old ← D^t_new;
end while
return F_1, F_2 , and all F_new generated in the step (3)
```

Given an unseen sample P^t in the tth task, MIML-Circle follows the principle of ensemble learning to predict the labels of P^t according to the outputs of all classifiers generated in each iteration for the tth task. Specifically, assume there are n classifiers generated. If more than $\theta \times n$ classifiers identify P^t as negative on a class label y, then the final predictive result about the label y is set to -1, and otherwise 1. Here $0 \leqslant \theta \leqslant 1$ denotes a decision threshold.

4 Experiments

4.1 Data preparation

To the best of our knowledge, this is the first attempt to use a multi-task MIML method to predict student performance. Since there are no public datasets available for our study, our experiments are based on self-collected datasets from a private higher education institution. The datasets are generated with the information about students' scores, syllabus, and course records. After data preprocessing and integration, five multi-task MIML datasets are generated, namely "Term2", "Term3", "Term4", "Term5", and "Term6", as shown in Table 2. For each dataset, we split it into two parts according to the chronological order of student registration, i.e., we take the latest grade students' information as testing set, and the data of the other older grade students as training set.

As shown in Table 2, the dataset "Term2" includes 1,020 student samples, who come from seven different computer-related specialties, such as "electronic technology", "computer science and technology", "computer network", and "computer information management". We view predicting student performance in different specialties as different tasks. Each task is associated with an MIML dataset. As traditional supervised learning methods utilize only one single vector to represent a sample, we also generate a groups of single-instance single-label (SISL) datasets. These SISL datasets contain merely score information of compulsory courses, and the information of most optional courses are discarded.

4.2 Evaluation metrics

As mentioned earlier, we view predicting student performance for each specialty as a single task, and each task has multiple courses to predict. In our study, we evaluate each algorithm in term of their average performance on all target courses, including average accuracy, average recall, average precision, and macro F_score [36]. For convenience, we denote these metrics as *ave_Acc*, *macro_Rec*, *macro_Prec*, and $macro\_F_\beta$, respectively. These metrics can be calculated as:

$$ave\_Acc = \frac{\sum_{t=1}^{m} \sum_{k=1}^{s^t} Acc^t_k}{\sum_{t=1}^{m} s^t}. \tag{2}$$

Here, m denotes the number of tasks (i.e., different special-

Table 2 Data description

Dataset	Training samples	Testing samples	Tasks	Training samples per task	Testing samples per task
Term2	1,020	147	7	67,92,125,215,96,246,179	19,12,18,19,25,29,25
Term3	969	147	7	67,92,111,215,59,246,179	19,12,18,19,25,29,25
Term4	676	115	7	67,92,48,94,38,234,103	19,12,7,19,21,12,25
Term5	253	50	3	67,92,94	19,12,19,
Term6	159	31	2	67,92	19,12

ties), and s^t is the number of target courses (i.e., labels) offered in the tth task; Acc_k^t denotes the prediction accuracy rate for the kth course in the tth specialty. Actually, ave_Acc gives the average accuracy of all courses offered in all related specialties. Similarly, $macro\_Rec$, $macro\_Prec$, and $macro\_F_\beta$ can be calculated as follows.

$$macro\_Rec = \frac{\sum_{t=1}^{m}\sum_{k=1}^{s^t} Rec_k^t}{\sum_{t=1}^{m} s^t}, \tag{3}$$

$$macro\_Prec = \frac{\sum_{t=1}^{m}\sum_{k=1}^{s^t} Prec_k^t}{\sum_{t=1}^{m} s^t}, \tag{4}$$

$$macro\_F_\beta = \frac{(1+\beta^2)\times macro\_Prec \times macro\_Rec}{(\beta^2 \times macro\_Prec + macro\_Rec)}, \tag{5}$$

where Rec_k^t and $Prec_k^t$ are the values of recall and precision on the kth course of the tth task, respectively. Given the kth course of the tth task, Rec_k^t is the fraction of the at-risk students that have been detected correctly by the model over the total amount of at-risk students. $Prec_k^t$ is the fraction of the at-risk students that have been detected correctly among all students identified as at-risk by the model. $macro\_F_\beta$ is a comprehensive metric of $macro\_Rec$ and $macro\_Prec$, in which β is larger than zero and measures the relative importance of the $macro\_Rec$ to $macro\_Prec$. Drawing from the experience in [8], we set the value of β to 1.5. We prefer the metric $macro\_Rec$, because that even if a course is falsely predicted to be difficult for a student, the final score of that student could also be improved owing to extra attention and guidance from teachers. Conversely, if we cannot detect a potential difficult course for the student, it is more likely that the student fails on the course.

4.3 Performance comparison

In order to verify the validity of MIML-Circle, we compare it with four other approaches, including MIMLSVM, SISL-Circle, the base classifier SVM, and the MIML method used in [17]. Here, SISL-Circle is a variant method of ours that adopts SISL instead of MIML as the backbone of the learning framework. All methods have been fully implemented in Matlab and tested on a PC equipped with 8-core 3.60GHz Intel Core processor and 16GB RAM. In this study, the iteration number R is set to 10, and the decision parameter θ is 0.6 on Term2, Term3, and Term4, and 0.8 on Term5 and Term6. The experimental results on five real datasets are reported in Table 3–Table 7, respectively.

Table 3 Performance of different algorithms on Term2

Methods	ave_Acc	$macro\_Rec$	$macro\_Prec$	$macro\_F_{1.5}$
SVM	0.7750	0.3654	0.3805	0.3700
MIMLSVM	0.7744	0.5004	0.4653	0.4890
SISL-Circle	0.7363	0.2323	0.2792	0.2449
The method in [17]	0.7177	0.5243	0.3420	0.4504
MIML-Circle	**0.8254**	**0.5893**	**0.5637**	**0.5811**

Table 4 Performance of different algorithms on Term3

Methods	ave_Acc	$macro\_Rec$	$macro\_Prec$	$macro\_F_{1.5}$
SVM	0.7807	0.3347	0.4802	0.3691
MIMLSVM	0.7676	0.4671	0.4358	0.4570
SISL-Circle	0.6817	0.5355	0.3259	0.4470
The method in [17]	0.7109	0.5200	0.3534	0.4541
MIML-Circle	**0.8297**	**0.6454**	**0.5836**	**0.6251**

Table 5 Performance of different algorithms on Term4

Methods	ave_Acc	$macro\_Rec$	$macro\_Prec$	$macro\_F_{1.5}$
SVM	0.7958	0.3237	0.4007	0.3441
MIMLSVM	0.7859	0.6187	0.5047	0.5785
SISL-Circle	0.6951	0.5596	0.3127	0.4502
The method in [17]	0.6855	0.5012	0.3165	0.4249
MIML-Circle	**0.8506**	**0.7038**	**0.5920**	**0.6651**

Table 6 Performance of different algorithms on Term5

Methods	ave_Acc	$macro\_Rec$	$macro\_Prec$	$macro\_F_{1.5}$
SVM	**0.7690**	0.1319	0.0972	0.1189
MIMLSVM	0.7059	0.5648	**0.3236**	0.4594
SISL-Circle	0.6613	0.3806	0.2062	0.3020
The method in [17]	0.6915	**0.7417**	0.3219	**0.5293**
MIML-Circle	0.6966	0.5981	0.3052	0.4618

Table 7 Performance of different algorithms on Term6

Methods	ave_Acc	$macro\_Rec$	$macro\_Prec$	$macro\_F_{1.5}$
SVM	0.7116	0.0833	0.0500	0.0691
MIMLSVM	0.5938	**0.7500**	0.3395	0.5466
SISL-Circle	0.6963	0.2708	0.2292	0.2565
The method in [17]	0.6705	0.5417	0.2896	0.4272
MIML-Circle	**0.7725**	0.6667	**0.4313**	**0.5708**

From Table 3–Table 7, we can see that MIML-Circle outperforms the other competitors. More precisely, it achieves the best performance on all of the four criteria on three datasets, i.e., Term2, Term3, and Term4. On Term5 and Term6, MIML-Circle is still competitive although the number of tasks are much less. On the dataset Term5, its performance is only inferior to the MIML method used in [17]. On the dataset Term6, it achieves the best performance on three criteria. In general, MIML-Circle obviously outperforms both MIMLSVM and SISL-Circle, which further illustrate the advantage of combining the multi-task learning and MIML learning.

4.4 Effect of the iteration number R

In order to understand the convergence of the algorithm MIML-Circle, we set the iteration number R to 20. Figure 2 shows its convergence on the five datasets in terms of $macro\_F_{1.5}$. It can be observed that all performance curves on different datasets have a similar variation trend. Specifically, as R increases, the performance curves go up rapidly at first, but when R is beyond a certain threshold, they maintains relatively stable with further increase of R. In our case, MIML-Circle achieves the best performance when $R = 12$ on most of the datasets.

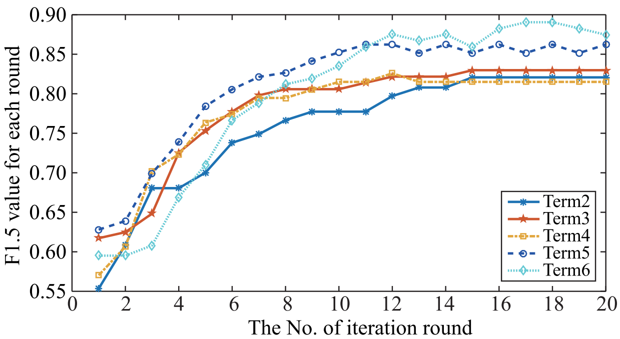

Fig. 2 The $macor\_F_{1.5}$ value per iteration

4.5 Effect of the decision parameter θ

When the training of MIML-Circle is completed, we estimate the label of a sample using the ensemble method as aforementioned. In this section, we study the influence of the decision threshold θ. Figure 3 shows how the change of θ affects the performance in terms of $macro\_F_{1.5}$.

From Fig. 3, we can observe that $macro\_F_{1.5}$ value improves remarkably on all the five datasets with the increase of the parameter θ, especially on the dataset Term4, Term5 and Term6. A possible reason is that the bigger the θ value, the more likely it is to estimate a sample to be a positive one, which leads to a higher recall value. It indicates that the ensemble mechanism plays an important role in detecting at-risk students, which is essential for the academic early warning system.

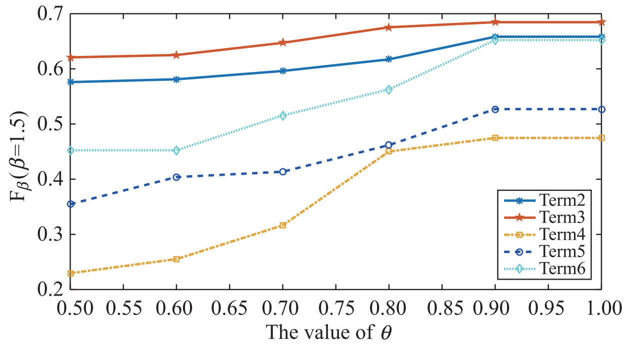

Fig. 3 The influence of θ in terms of $macro\_F_{1.5}$

5 Discussion and conclusions

In this paper, we focus on traditional classroom-teaching scenes and predict student performance prior to the commencement of new courses. With this technique, some assistant teaching means can be conducted during the initial learning period of a new course, which can facilitate the studying in the follow-up stages. We cast the problem as an MIML learning problem, which leverages not only the inconsistent course information across different students, but also the correlations among target courses. In addition, we collect five real data sets for pre-course student performance prediction and propose a novel multi-task learning method to alleviate the data scarcity problem. Experimental results have demonstrated the promise of our method for pre-course student performance prediction in comparison with traditional approaches.

It should be noted that there exist many other factors affecting student performance, such as psychological status, family, and health. Moreover, student learning behavior in each semester is not exactly the same, which makes the task of student performance prediction very challenging. Thus, it is highly appealing to consider more factors to predict student performance in the future.

Acknowledgements This work was supported by the National Natural Science Foundation of China (Grant Nos. 61701281, 61573219, and 61876098), Shandong Provincial Natural Science Foundation (ZR2016FM34 and ZR2017QF009), Shandong Science and Technology Development Plan (J18KA375), Shandong Social Science Project (18BJYJ04), and the Fostering Project of Dominant Discipline and Talent Team of Shandong Province Higher Education Institutions.

References

1. Sweeney M, Rangwala H, Lester J, Johri A. Next-term student perfor-

mance prediction: a recommender systems approach. Journal of Educational Data Mining, 2016, 8(1): 22–51

2. Grayson A, Miller H, Clarke D D. Identifying barriers to help-seeking: a qualitative analysis of students' preparedness to seek help from tutors. British Journal of Guidance & Counselling, 1998, 26(2): 237–253
3. Romero C, Ventura S. Educational data mining: a review of the state of the art. IEEE Transactions on Systems Man and Cybernetics, Part C (Application and Reviews), 2010, 40(6): 601–618
4. Qiujie L, Rachel B. The different relationships between engagement and outcomes across participant subgroups in massive open online courses. Computers & Education, 2018, 127: 41–65
5. Ren Z, Rangwala H, Johri A. Predicting performance on MOOC assessments using multi-regression models. In: Proceedings of the 9th International Conference on Education Data Mining. 2016, 484–489
6. Trivedi S, Pardos Z A, Heffernan N T. Clustering students to generate an ensemble to improve standard test score predictions. In: Proceedings of International Conference on Artificial Intelligence in Education. 2011, 377–384
7. Er E. Identifying at-risk students using machine learning techniques: a case study with is 100. International Journal of Machine Learning and Computing, 2012, 2(4): 476–480
8. Hu Y H, Lo C L, Shih S P. Developing early warning systems to predict students online learning performance. Computers in Human Behavior, 2014, 36: 469–478
9. Macfadyen L P, Dawson S. Mining LMS data to develop an early warning system for educators: a proof of concept. Computers & Education, 2010, 54(2): 588–599
10. Zafra A, Romero C, Ventura S. Multiple instance learning for classifying students in learning management systems. Expert Systems with Applications, 2011, 38(12): 15020–15031
11. Kotsiantis S B, Pierrakeas C J, Pintelas P E. Preventing student dropout in distance learning using machine learning techniques. Applied Artificial Intelligence, 2004, 18(5): 411–426
12. Xenos M. Prediction and assessment of student behaviour in open and distance education in computers using bayesian networks. Computers & Education, 2004, 43(4): 345–359
13. Marbouti F, Diefes-Dux H A, Madhavan K. Models for early prediction of at-risk students in a course using standards-based grading. Computers & Education, 2016, 103: 1–15
14. Meier Y, Xu J, Atan O, Schaar M V D. Predicting grades. IEEE Transactions on Signal Processing, 2016, 64(4): 959–972
15. Gedeon T D, Turner S. Explaining student grades predicted by a neural network. In: Proceedings of International Joint Conference on Neural Networks. 2002, 609–612
16. Acharya A, Sinha D. Early prediction of students performance using machine learning techniques. International Journal of Computer Applications, 2014, 107(1): 37–43
17. Ma Y L, Cui C R, Nie X S, Yang G P, Shaheed K, Yin Y L. Pre-course student performance prediction with multi-instance multi-label learning. Science China Information Sciences, 2019, 62(2): 200–205
18. Shalevshwartz S, Bendavid S. Understanding Machine Learning. 1st ed. New York: Cambridge University Press, 2014
19. Zhou Z H, Zhang M L. Multi-instance multi-label learning with application to scene classification. In: Proceedings of the 19th International Conference on Neural Information Processing Systems. 2006, 1609–1616
20. Zhang Y, Yang Q. A survey on multi-task learning. 2017, arXiv preprint arXiv:1707.08114
21. Wang A Y, Newlin M H, Tucker T L. A discourse analysis of online classroom chats: predictors of cyber-student performance. Teaching of Psychology, 2001, 28(3): 222–226
22. Wang A Y, Newlin M H. Predictors of performance in the virtual classroom: identifying and helping at-risk cyber-students. Journal of Higher Education Academic Matters, 2002, 29(10): 21–25
23. Essa A, Ayad H. Student success system: risk analytics and data visualization using ensembles of predictive models. In: Proceedings of International Conference on Learning Analytics and Knowledge. 2012, 158–161
24. Lopez M I, Luna J M, Romero C, Ventura S. Classification via clustering for predicting final marks based on student participation in forums. In: Proceedings of International Conference on Educational Data Mining. 2012, 148–151
25. Zhang M L, Zhou Z H. M3MIML: a maximum margin method for multi-instance multi-label learning. In: Proceedings of the 8th International Conference on Data Mining. 2008, 688–697
26. Zhang M L. A k-nearest neighbor based multi-instance multi-label learning algorithm. In: Proceedings of the 22nd International Conference on Tools with Artificial Intelligence. 2010, 207–212
27. Xu X S, Xue X, Zhou Z H. Ensemble multi-instance multi-label learning approach for video annotation task. In: Proceedings of the 19th International Conference on Multimedea. 2011, 1153–1156
28. Li Y F, Hu J H, Jiang Y, Zhou Z H. Towards discovering what patterns trigger what labels. In: Proceedings of the 26th AAAI Conference on Artificial Intelligence. 2012, 1012–1018
29. Huang S J, Zhou Z H. Fast multi-instance multi-label learning. In: Proceedings of the 28th AAAI Conference on Artificial Intelligence. 2014, 1868–1874
30. Feng J, Zhou Z H. Deep MIML network. In: Proceedings of the 31st AAAI Conference on Artificial Intelligence. 2017, 158–161
31. Yang Y, Wu Y F, Zhan D C, Liu Z B, Jiang Y. Complex object classification: a multi-modal multi-instance multi-label deep network with optimal transport. In: Proceedings of the 24th ACM International Conference on Knowledge Discovery and Data Mining. 2018, 2594–2603
32. Zhou Z H, Zhang M L. Solving multi-instance problems with classifier ensemble based on constructive clustering. Knowledge & Information Systems, 2007, 11(2): 155–170
33. Boutell M R, Luo J, Shen X, Brown C M. Learning multi-label scene classification. Pattern Recognition, 2004, 37(9): 1757–1771
34. Zhou Z H. Ensemble Methods: Foundations and Algorithms. 1st ed. Florida: CRC Press, 2012
35. Wang S B, Li Y F. Classifier circle method for multi-label learning. Journal of Software, 2015, 26: 2811–2819
36. Zhou Z H. Machine Learning. 1st ed. Beijing: Tsinghua University Press, 2016

Yuling Ma received the Master degree in computer science and technology from Shandong University, China in 2008. She is currently pursuing her PhD degree at Shandong University, China. Her research interests are machine learning and data mining, and educational data mining, with a current specific focus on student performance prediction.

Chaoran Cui received his PhD degree in computer science from Shandong University, China in 2015. Prior to that, he received his BE degree in software engineering from Shandong University, China in 2010. During 2015–2016, he was a research fellow at Singapore Management University, Singapore. He is now a professor with School of Computer Science and Technology, Shandong University of Finance and Economics, China. His research interests include information retrieval, recommender systems, multimedia, and machine learning.

Jun Yu received the Bachelor degree in software engineering from Shandong University, China in 2017. He is currently researching machine learning and data mining in MLA laboratory as a postgraduate student at Shandong University, China. His research interests are computer vision and deep learning, with a specific focus on image analysis and understanding.

Jie Guo received the Master degree in School of Information Science and Engineering from Shandong Normal University, China in 2015. She is currently pursuing her PhD degree at Shandong University, China. Her research interests are machine learning and multimedia analysis.

Gongping Yang received his PhD degree in computer software and theory from Shandong University, China in 2007. Now he is a professor in the School of Software Engineering, Shandong University, China. His research interests are pattern recognition, image processing, biometrics, and so forth.

Yilong Yin is a professor in School of Software Engineering and the director of the MLA Lab. He received his PhD degree from Jilin University, China in 2000. From 2000 to 2002, he worked as a post-doctoral fellow in the Department of Electronic Science and Engineering, Nanjing University, China. His research interests are machine learning and data mining, computational medicine, and biometrics.