So, H. J. et al. (Eds.) (2020). Proceedings of the 28th International Conference on Computers in Education. Asia-Pacific Society for Computers in Education

Predicting end-of-session actions considering the information of learning materials

Daichi Takehara[a*]
[a]*Aidemy Inc., Japan*
*takehara-d@aidemy.co.jp

Abstract: To provide learners with a better learning experience in online educational systems, it is meaningful to understand and model learners' actions. The actions of learners ending their learning sessions and leaving the systems, which we denote as the end-of-session actions, is important to understand. Modeling the end-of-session actions can lead to useful applications, such as optimizing the way learning materials are presented and interventions that can appropriately help learners. This paper addresses the problem of predicting end-of-session actions in online educational systems. While previous studies have mainly focused on the learners' behavior in the systems, this paper focuses on incorporating the information of learning materials into the prediction model. Learning material features were extracted by considering multiple perspectives in the learning materials, including their order in the course and their texts. The experiment was conducted using actual user log data from the programming learning system. The experiment demonstrates the effectiveness of incorporating learning material features into the prediction models and analyzed their contribution to the prediction accuracy.

Keywords: end-of-session action prediction, learning materials modeling, feature extraction, machine learning

1. Introduction

Recently, online educational systems, such as Massive Open Online Courses (MOOCs), have been established and widely used. In many cases, online educational systems allow learners to actively choose the time to study and the order of learning materials to be taken, depending on their learning style. Learners voluntarily access the system to begin and end their learning. Given that learners use the system in this way, it is worthwhile to model the actions of learners ending their learning sessions and leaving the systems, which we denote as end-of-session actions. By being able to model these accurately, systems can provide a better learning experience for learners through optimizing the way learning materials are presented and interventions that can appropriately help learners.

Several studies have addressed the problem of modeling and predicting end-of-session actions of learners in online educational systems (Karumbaiah et al., 2018; Hansen et al., 2019). Karumbaiah et al. (2018) proposed a method that predicted the end of students playing a learning game called Physics Playground developed for physics education for middle school students. The authors designed 101 features, focusing on progress and experiences based on learners' behavioral history in the game. The authors used the obtained features in a prediction model with a gradient-boosting tree. Hansen et al. (2019) presented the problem of determining when a session ends by modeling the probability that an action will be the end of the session. Through modeling by long short-term memory (LSTM) (Hochreiter et al., 1997), the authors showed the effectiveness of considering learners' long-term behavioral history in the problem. Although these previous studies paid attention to learners' behavior in the educational systems, insufficient consideration was given to incorporating the information of the learning materials into the model.

This paper addresses the prediction of end-of-session actions incorporating the information of learning materials. This paper proposes a model that predicts the probability if the learner's answer to an exercise will be the end-of-session action. The proposed method extracts not only the features related to the learners' actions, i.e., the learner features, but also the features related to the learning materials, i.e., the learning material features, from the log data stored in the system. The learning material features are extracted by considering multiple perspectives in the learning materials, including their order in the course and their texts. Based on the extracted features, the proposed method learns the prediction model by machine learning, linear regression (Seber et al., 2012), random forest (Breiman, 2001), and gradient-boosting tree (Friedman, 2001) algorithms. The experiment was conducted using actual learners' log data on Aidemy, an online programming learning system (the system is described in detail in section 2.1). The experiment verified the effectiveness of incorporating the learning material features into the prediction of end-of-session actions and analyze their contribution to the prediction accuracy of them in detail.

2. Data

2.1 Aidemy

In this paper, we used the log data of learners on Aidemy. Aidemy is an online programming learning system that allows learners to acquire skills and knowledge about data science, such as statistics and machine learning. The main feature is that learners can learn not only by reading learning materials, but also by coding in the browser editor, as shown in Figure 1. Each course includes a technical topic (e.g., "Introduction to Machine Learning," "Introduction to Python," and "Fundamentals of Deep Learning") and consists of multiple exercises. The types of exercises include multiple choice questions, coding questions, and videos. After registering for the system, learners can purchase and start taking courses that interest them from the 46 courses published as of December 20, 2019.

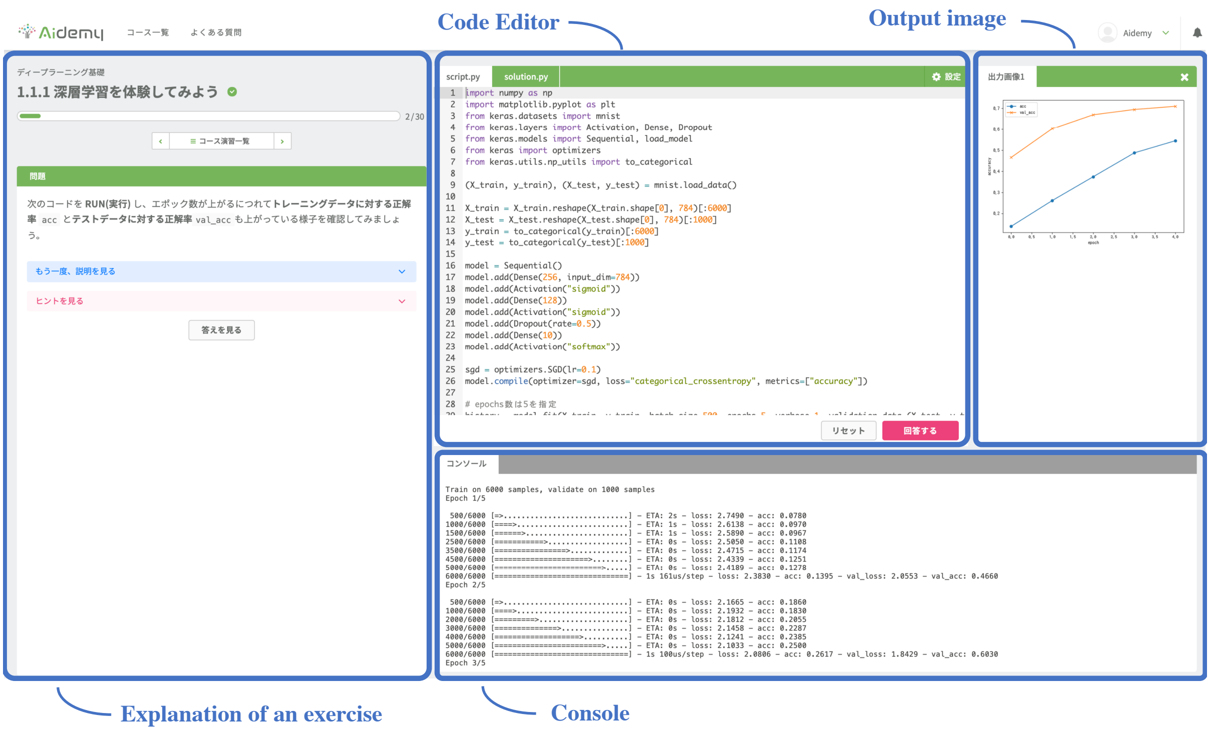

Figure 1. The screen where a learner is taking the exercise in Aidemy. The left side of the screen shows the explanation of the exercises, and the right side of the screen shows questions that are answered by inputting the code or the answer to the multiple questions.

2.2 Log Data

The log data used in the experiment is generated when a learner answers an exercise on Aidemy. The log data includes the timestamp, the learner id, the course id, and the exercise id. The course id and the exercise id are associated with information about each learning material, including the title and description of the learning material, the type of exercise, and the Python libraries used. In this paper, the proposed method extracts learner features and learning material features from these log data and uses them for training a predictive model.

We define end-of-session actions by the length of time spent on the answer based on the timestamp of the log data. If the length of the answer is more than 15 minutes, the action is considered to be the end-of-session action. Since the length of time spent on the answer of a single exercise on Aidemy often ranges from a few tens of seconds to a few minutes and rarely exceeds 15 minutes, we empirically set the threshold value to 15 minutes.

In the experiment, we collected the log data for the period from January 1, 2019, to November 30, 2019. Learners who had purchased one or more paid courses and completed at least ten exercises were used for the experiment. The collected log data included 1314 learners, 326890 answers, and 35033 end-of-session actions.

3. Predicting End-of-session Actions Considering the Information of Learning Materials

3.1 Problem Setting

This paper addresses the problem of predicting end-of-session actions, as in the paper by Hansen et al. (2019). In this problem, the prediction model outputs the probability if the learner's answer to the exercise will be an end-of-session action. The training dataset used pairs of a feature extracted from the learner's answer and a binary label.

3.2 Feature Extraction

The features extracted from the log data are described in Table 1. The features are divided into multiple separate feature groups, learner features, and four learning material features (basic, order, text, and library). Learning material features are divided into subgroups based on their characteristics for more detailed analysis in the experiment.

Table 1. *Overview of Features*

Group	Feature name	Description
Learner	learner id	learner identifier
	session num	number of sessions
	stay time in last answer	answering time in the previous answer
	stay time from last session	total answering time since the previous session
	session length	number of answers in the current session
	mean session length	mean of the number of answers in the learner's session
	std session length	standard deviation of the number of answers in the learner's session
	answer time zone	answer time zone (6 different time zones of the day, 4 hours each for the 24 hours in a day)
	learner time zone	ratio of answer time zone of the learner
Learning material (basic)	course id	course identifier
	exercise id	exercise identifier
	exercise type	types of exercises (choice question, code question, video)
	change exercise	indicator corresponding to whether the exercise is different from the previous one or not
	change course	indicator corresponding to whether the course is different from the previous one or not
Learning material (order)	order	order of the exercise (counting from the beginning of the course)
	order from chapter end	order of the exercise (counting from the end of the chapter)

	order from end	order of the exercise (counting from the end of the course)
Learning material (text)	text feature	dense features (5 dimensions), which are extracted by applying the term frequency-inverse document frequency (TFIDF) (Manning et al., 2008) to course and exercise titles and descriptions and and dimensionality reduction by singular value decomposition (SVD) (Halko et al., 2009)
Learning material (library)	library	categorical features of which Python library is being used in each course. It targets the libraries that are being imported by the code input exercises. For example, they include NumPy, pandas, and scikit-learn, etc.

The learner features are extracted based on the history of learners on the systems, which also represent the learners' behavior and recent attendance.

Learning material features are extracted by considering multiple perspectives of the learning materials. Learning material (basic) contains basic information, such as id and types of learning materials. Learning material (order) contains the information about the order of the exercises in the course. It may contain some meaning depending on the order of the exercises, e.g., exercises at the end of a course or the end of a chapter are likely to be the timing of a learning break. Learning material (text) is extracted by using the text contained in the learning materials and represents their content. Leaching materials (libraries) represent the Python libraries used in the course. This information is essential for learners of the programming learning system.

3.3 Models

Based on the extracted features, the proposed method learns the prediction model by machine learning algorithms. To learn the prediction model, we used three algorithms: linear regression (Seber et al., 2012), random forest (Breiman, 2001), and gradient-boosting tree (Friedman, 2001) algorithms in the experiment. Linear regression is a simple algorithm that assumes a linear relationship between features and targets. Decision tree-based algorithms, such as random forest and gradient-boosting trees, are advanced algorithms that can effectively learn features. Moreover, a detailed analysis using the importance of the features with Gini importance (Breiman, 2001) can be performed. This analysis is consistent with the primary purpose of this paper, which is to verify the effectiveness of each learning material feature.

4. Experiments

In the experiment, we verified the effectiveness of incorporating the information of learning materials into the prediction of end-of-session actions by using actual log data in Aidemy.

4.1 Experimental setup

The evaluation was performed using the time-series cross-validation approach (Hyndman, 2019), which divided the log data into multiple time series. Five datasets were created in the experiment. Each dataset consisted of a 6-month training dataset and a test dataset for the following month (e.g., if the training dataset was collected between January and June 2019, then the test dataset was collected in July 2019.). The evaluation scores were then calculated by averaging the score for each test dataset. Validating the model over multiple periods in this way increases the reliability of the evaluated scores. Five-month test datasets for July, August, September, October, and November 2019, were used in the experiment. The evaluation metric was AUC, which is the area under the receiver operating characteristic (ROC) curve.

The prediction model was trained using multiple combinations of the features described in Section 3.2. Specifically, we compared the use of only the learner features (**Learner**), the use of one of the learning material features in addition to the learner features (**Learner w/ learning material feature group**), and the use of all the features (**All features**). The prediction models were trained with the linear regression, random forest, and gradient-boosting tree algorithms, as described in 3.3. These algorithms use an implementation of the Python library, Scikit-learn (Pedregosa et al., 2011), and the hyperparameters are used as their default values.

4.2 Results

The experimental results are shown in Table 2. The AUCs of the cases when incorporating learning material features (All features, Learner w/ learning material feature group) were higher than cases when using only learner features (Learner). As a result, the effectiveness of incorporating the learning material features in predicting end-of-session actions was confirmed.

Table 2. *Evaluation Results.*

Algorithm	**Feature***	**AUC**
Linear Regression	Learner	0.6250
	Learner w/ Learning Material (basic)	0.6369
	Learner w/ Learning Material (order)	0.6251
	Learner w/ Learning Material (text)	0.6540
	Learner w/ Learning Material (library)	0.6393
	All features	**0.6651**
Random Forest	Learner	0.6078
	Learner w/ Learning Material (basic)	0.6621
	Learner w/ Learning Material (order)	0.6661
	Learner w/ Learning Material (text)	0.6725
	Learner w/ Learning Material (library)	0.6137
	All features	**0.6840**
Gradient-boosting tree	Learner	0.6650
	Learner w/ Learning Material (basic)	0.6928
	Learner w/ Learning Material (order)	0.7067
	Learner w/ Learning Material (text)	0.7073
	Learner w/ Learning Material (library)	0.6697
	All features	**0.7155**

*The notation of the feature columns corresponds to the group columns in Table 1.

4.3 Further Analysis

The experimental results show that, although all learning material features contribute to improve the prediction accuracy, the degree of their contribution is different. To confirm how each feature works in more detail, we observed the Gini importance (Breiman, 2001) in the gradient-boosting tree as the feature importance. Figure 2 shows the feature importance aggregated by the sum of each feature group shown in Table 1 (left figure) and the top ten features of the higher feature importance (right figure).

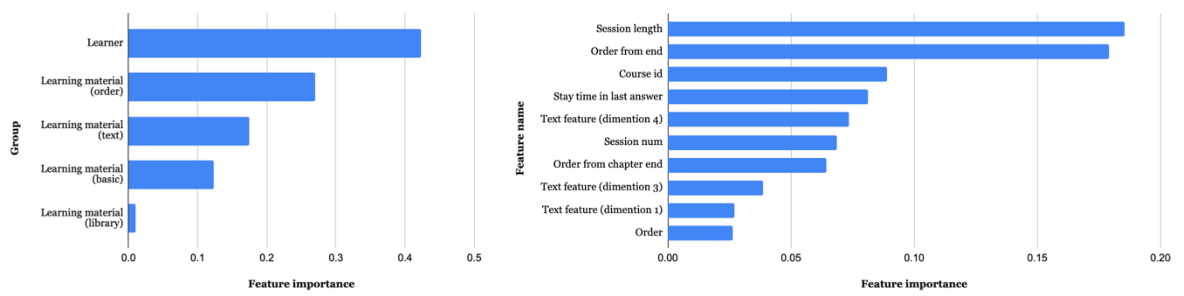

Figure 2. Feature importance (left: the feature importance aggregated by the sum of each feature group, right: the top ten features of the higher feature importance). The notation of the vertical axis corresponds to the group and feature columns in Table 1.

Similarly to the degree of improvement in AUCs in Table 2, the importance of Learning material (order) and Learning material (text) was confirmed to be high. Learning material (order) is consistent with the intuitive understanding that delimitations such as the end of the course and the end of the chapter in a course are strongly related to whether learners continue learning or not. In addition, Learning material (text) is a feature that represents semantics about the content of the learning material, such as a technical topic. Therefore, it is suggested that the content of the learning material had a moderate effect on the learner's end-of-session actions.

5. Conclusions

This paper addressed the problem of predicting end-of-session actions incorporating the information of learning materials. The proposed method extracted not only the learner features, but also the learning material features from the log data stored in the system. Based on the extracted features, the proposed method learned the prediction model by machine learning algorithms.

In the experiment using actual log data in Aidemy, the effectiveness of incorporating the learning material features into the prediction of end-of-session actions was confirmed. In addition, the features' importance and contribution to the predictive model was verified. In this experiment, we can confirm that features related to the order of the exercises in the course and text contained in the learning materials are significant contributors to prediction accuracy.

References

Breiman, L. (2001) Random forests, *Machine Learning*, *45*(1), 5-32.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189-1232.

Halko, N., Martinsson, P. G., & Tropp, J. A. (2009) Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, *53*(2), 217-288.

Hansen, C., Hansen, C., Alstrup, S., & Lioma, C. (2019). Modeling end-of-session actions in educational systems. In *Proceedings of the International Conference on Educational Data Mining*, (pp. 306-311).

Hochreiter, S. & Schmidhuber, J. (1997) Long short-term memory, *Neural Computation*, *9*(8), 1735-1780.

Hyndman, R. J. & Athanasopoulos, G. (2019) Forecasting: Principles and Practice (3rd edition), Melbourne: OTexts. https://otexts.com/fpp3/

Karumbaiah, S., Baker, R. S., & Shute, V. (2018) Predicting quitting in students playing a learning game. In *Proceedings of the International Conference on Educational Data Mining*, (pp. 167-176).

Manning, C., Raghavan, P., & Schütze, H. (2010). Introduction to information retrieval. *Natural Language Engineering*, *16*(1), 100-103

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011) Scikit-learn: Machine learning in Python, *Journal of Machine Learning Research*, *12*, 2825-2830.

Seber, G. A., & Lee, A. J. (2012). *Linear regression analysis* (Vol. 329). John Wiley & Sons.