

Dropout Prediction over Weeks in MOOCs by Learning Representations of Clicks and Videos

Byungsoo Jeon^{1*}, Namyong Park^{1*}

¹School of Computer Science, Carnegie Mellon University, Pittsburgh, PA

¹{byungsoj,namyongp}@cs.cmu.edu

Abstract

This paper addresses a key challenge in MOOC dropout prediction, namely to build meaningful representations from clickstream data. While a variety of feature extraction techniques have been explored extensively for such purposes, to our knowledge, no prior works have explored modeling of educational content (e.g. video) and their correlation with the learner’s behavior (e.g. clickstream) in this context. We bridge this gap by devising a method to learn representation for videos and the correlation between videos and clicks. The results indicate that modeling videos and their correlation with clicks brings statistically significant improvements in predicting dropout.

1 Introduction

One of the most important challenges in MOOCs is to identify students most at risk with the goal of providing supportive interventions. Considering the huge number of users taking courses in MOOCs, it is necessary to automate the at-risk identification process. While there are several data sources to leverage in MOOCs, such as forum posts, assignments, and clickstreams, a recent study (Gardner and Brooks 2018) proves that clickstream-based features are significantly better in dropout prediction. However, the raw clickstream data are too fine-grained to build a meaningful representation for downstream classification models, e.g. a dropout prediction model.

Many researchers have offered techniques to transform the raw clickstream data into the structured feature representation acceptable to existing statistical or machine learning models. One way is to use hand-crafted features, e.g. the count of each click type (Halawa, Greene, and Mitchell 2014; Lykourantzou et al. 2009; Yang et al. 2013; Nagrecha, Dillon, and Chawla 2017; Whitehill et al. 2015). However, some studies use raw clickstream data without this feature engineering, claiming that it removes an important sequential pattern of the clickstream (Fei and Yeung 2015; Whitehill et al. 2017; Wang, Yu, and Miao 2017; Kim, Vizitei, and Ganapathi 2018). Still, these end-to-end

models may overlook important patterns in the data by taking only a single objective into account. This naturally leads researchers to consider unsupervised methods to capture meaningful patterns under multiple objectives (Sinha et al. 2014a; 2014b).

On top of this line of work, we hypothesize that the modeling of educational content (e.g. video) and their correlation with the learner’s behavior (e.g. clickstream) should help dropout prediction. For instance, let’s assume we observe a sequence of clicks indicating the struggles of learners (e.g. stopping the video). However, the same click pattern should be interpreted in a different way depending on the difficulty of educational contents. Nevertheless, no studies have been done in this direction. Thus, to test our hypothesis, we present dropout prediction model to explicitly capture the correlation between clicks and videos. We also propose a method to learn representation for clicks and videos in an unsupervised fashion. Our experimental results prove the benefit of modeling the correlation between clicks and videos and pretrained embeddings for click n-grams.

2 Related Work

A substantial number of studies have predicted how likely a user is to drop out on a variety of MOOC datasets. However, many studies are based upon easily interpretable and static features such as grade (Halawa, Greene, and Mitchell 2014; Lykourantzou et al. 2009; Whitehill et al. 2015), demographics (Lykourantzou et al. 2009), forum posts (Yang et al. 2013), etc (Yang et al. 2013). Nevertheless, the majority of user interaction data in MOOCs is in the form of hardly interpretable clickstreams, and clickstream-based features are recently proved to be superior to other features in dropout prediction (Gardner and Brooks 2018). Our work aims to utilize rich and valuable information on user states from clickstreams, which could explain the chance of dropout.

A few works have still made use of clickstreams to predict dropout. One of main challenges for using clickstream data is to convert clickstream data into fixed-length and meaningful representation for downstream classification models. The prevalent solution to this challenge is to use hand-crafted features extracted from clickstream (Li et al. 2016; Amnuey-pornsakul, Bhat, and Chinprutthiwong 2014; Taylor, Veera-

*Equal contribution

machaneni, and O’Reilly 2014; Kloft et al. 2014). These features include forum-related variables (e.g. the number of forum views), assignment-related variables (e.g. the number of submissions), and activity-related variables (e.g. the number of page/video views) (Nagrecha, Dillon, and Chawla 2017; Whitehill et al. 2015). However, these features inevitably lose temporal patterns in clickstreams and introduce unintentional biases due to the researchers’ subjectivity. On top of that, some of the hand-engineered features are not platform-agnostic, which requires us to modify the feature extraction methods depending on the platform.

To resolve these issues, some researchers have attempted to predict dropout without manual feature engineering processes. They adopt deep neural network models since these are proved to be surpassing feature extractors regardless of domains even if they take the data in its raw form as an input. Given the sequential nature of clickstream, sequence models, such as recurrent neural network (RNN) and long short-term memory (LSTM), are the most popular choices (Fei and Yeung 2015; Whitehill et al. 2017), while some researchers use more sophisticated deep neural network models (Wang, Yu, and Miao 2017; Kim, Vizitei, and Ganapathi 2018). While these end-to-end models take the human out of the loop, they often lose the important signals from the data by optimizing only a single objective (Glasmachers 2017). In contrast, our representation learning framework complements end-to-end models by building meaningful representations for click sequence in an unsupervised manner while preserving temporal information in clicks.

The most similar works to ours in this regard focus on constructing cognitively meaningful representations from the clickstream. One approach is to build a combined representation of clickstream and discussion forum footprints using a set of graph metrics (Sinha et al. 2014b). Another recent work groups raw click sequences into predefined behavioral categories and measures the degree of information processing as a proxy for the concentration (Sinha et al. 2014a). However, unlike our method, none of them explicitly models the correlation between clickstream and video. Moreover, both of them rely on the hand-wavy design of behavioral category or taxonomy that provides us with the interpretability at the cost of missing important temporal signals.

3 Proposed Method

In this section, we describe our proposed network for dropout prediction capturing the correlation between learning contents (video) and learner’s behavior (click sequence). Then, we present two approaches for learning click n-grams and video representation in an unsupervised fashion.

Dropout Prediction Network

Given a clickstream data, we model how a learner’s state evolves week by week until he completes the course, or drops out of it. Our intuition for the proposed network is that whether a student drops out or not is strongly associated with (1) his learning experiences or activities, and (2) the characteristics of the learning material. For instance, a student’s current understanding of the course topics, how effective a student’s learning strategy has been, how difficult

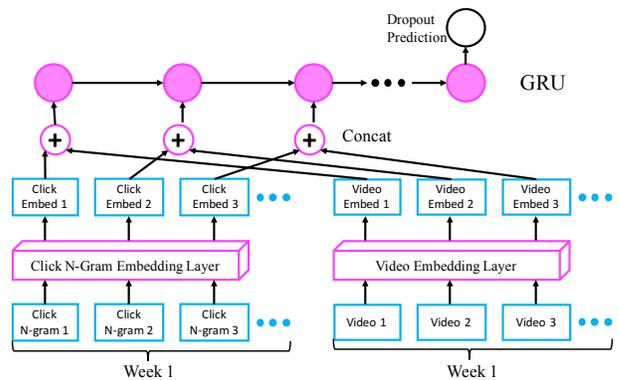


Figure 1: Dropout prediction network.

the learning material is, and how well the content is presented could all play a role in the decision of dropping out of a course.

In order to model a student’s learning activity, our model considers the sequential relationship among clicks. More specifically, our model uses a click n -gram, instead of a single click, as a unit for modeling a user activity based on the intuition that a single click is far too fine-grained, and a user activity could be better represented by a click sequence. We define a click n -gram to be a sequence of n consecutive clicks where each click c corresponds to one of ten predefined user clicks (see Section 4 for details). Our model also considers the characteristics of the learning material by receiving video information through a dedicated layer. Further, we model how the two closely-related factors, which we capture by way of click n -grams and videos, affect the student’s learning experience over time by combining them.

Figure 1 depicts our proposed network. Given a click stream for the current week, we generate a sequence of click n -grams and the corresponding video ids, and transform them into click n -gram and video representations via two separate embedding layers. Click n -gram and video embeddings are then concatenated to be fed into a gated recurrent unit (GRU) layer, which learns the transition of a student state for the current week, and outputs whether a student will drop out within the next week. Note that two embedding layers can be initialized with the weights pretrained by our unsupervised pretraining methods that follows.

To train our model, we minimize the margin ranking loss due to the imbalanced labels. We pair one positive instance (dropout) of the user with other negative instances of the same user. The objective function is

$$\min_{\mathbf{W}} \frac{1}{T} \sum_{t=1}^T \max(0, -(P_{pos} - P_{neg}) + M) \quad (1)$$

where \mathbf{W} is composed of all the model parameters; T is the number of pairs; P_{pos} and P_{neg} are the probability of dropout for positive and negative instance computed from our network; and M is margin, which is set to 0.5.

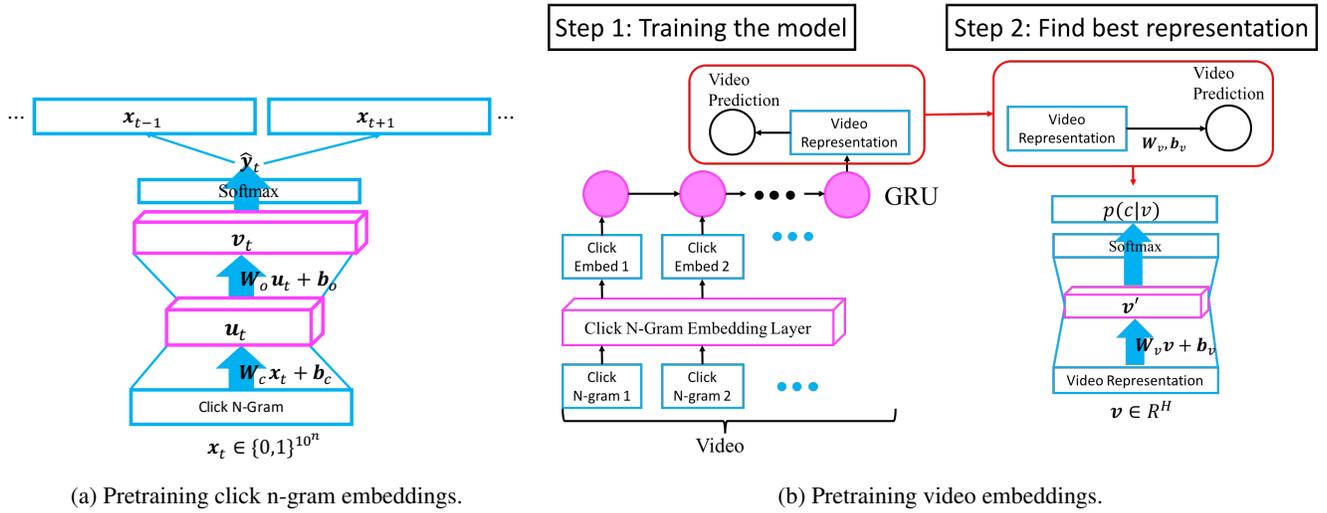


Figure 2: Pretraining embeddings of click n-grams and videos.

Pretraining Click N-Gram Embeddings

Given a sequence of clicks, we aim to capture the latent relationships between learners' consecutive clicks via learning a representation of the clickstream that reflects the inter-click sequential information. We train a multi-layer perceptron (MLP) using the following intuition: Learning is a continuous process for each student. Thus a representation corresponding to a student's activity at some point should be able to predict the student's activities in both the recent past and the near future.

Based on this intuition, we train the MLP architecture shown in Figure 2a. A click n -gram is represented as a one-hot vector of length 10^n . Given a click n -gram $\mathbf{x}_t \in \{0, 1\}^{10^n}$ for week t , the first layer transforms it into a latent representation $\mathbf{u}_t \in \mathbb{R}^m$ as follows:

$$\mathbf{u}_t = \mathbf{W}_c \mathbf{x}_t + \mathbf{b}_c$$

where $\mathbf{W}_c \in \mathbb{R}^{m \times 10^n}$ is the weight matrix for encoding a click n -gram, and $\mathbf{b}_c \in \mathbb{R}^m$ is the bias vector. Then the second layer constructs a vector $\mathbf{v}_t \in \mathbb{R}^{10^n}$ which is of the same size as the input \mathbf{x}_t :

$$\mathbf{v}_t = \mathbf{W}_o \mathbf{u}_t + \mathbf{b}_o$$

where $\mathbf{W}_o \in \mathbb{R}^{10^n \times m}$ is the weight matrix for decoding an encoded click n -gram, and $\mathbf{b}_o \in \mathbb{R}^{10^n}$ is the bias vector. Then, given the reconstructed representation \mathbf{v}_t , we minimize a cross entropy error as follows:

$$\min_{\substack{\mathbf{W}_c, \mathbf{W}_o, T \\ \mathbf{b}_c, \mathbf{b}_o}} \frac{1}{T} \sum_{t=1}^T \sum_{\substack{-w \leq i \leq w \\ (i \neq 0)}} -\mathbf{x}_{t+i}^\top \log \hat{\mathbf{y}}_t - (\mathbf{1} - \mathbf{x}_{t+i})^\top \log (\mathbf{1} - \hat{\mathbf{y}}_t)$$

where $\hat{\mathbf{y}}_t = \frac{\exp(\mathbf{v}_t)}{\sum_{j=1}^{10^n} \exp(\mathbf{W}_o[j, :] \mathbf{u}_t + \mathbf{b}_o[j])}$, w is context

window size, T is the total number of weeks, $\mathbf{1}$ is an all-one vector, and \exp is the element-wise exponential function.

Pretraining Video Embeddings

In order to pretrain video embeddings, we have two steps: training the pretraining model and finding the best representation vector for each video. For the first step, we build up and train the model to predict the video label given consecutive click n -gram sequence for that video as in Figure 2b. We use pretrained click embeddings and GRU to construct final video representation, which is the last hidden state of GRU. Based on this final video representation, our model makes a prediction on video label for given click n -gram sequence.

In the second step, our goal is to find the video representation v , which maximizes the probability of each video label given trained weight and bias of the last layer of this model, \mathbf{W}_v and \mathbf{b}_v . Let $v_{max}^{(i)}$ be the best video representation for video i and c be the true video label, then it is expressed as follows:

$$v_{max}^{(i)} = \operatorname{argmax}_v p(c = i | v)$$

$$\text{where } p(c = i | v) = \operatorname{softmax}(\mathbf{W}_v v + \mathbf{b}_v)_i$$

We can easily get $v_{max}^{(i)}$ by the family of gradient descent algorithms and use it as the embedding vector of video i in our dropout prediction network.

4 Experiment

Dataset

The dataset we use is collected from the Coursera¹, the top-ranked MOOC platform with more than 28 million users and 2,000 online courses, and is also used in (Yang et al. 2015). The dataset includes clickstream data that contain clicks of Coursera learners who took video lectures of the Microeconomics course for maximum 12 weeks. It includes 2,709,053 clicks collected from 48,090 users. Clicks are divided into 10 categories: *Pageview*, *Quiz*, *Forum*, *Play*, *Pause*, *SeekFwd*, *SeekBwd*, *RateFaster*, *RateSlower*, and *Stalled*.

¹<https://www.coursera.org/>

Table 1: Dropout prediction performance of models that use different types of information and pretrained embeddings.

Model	AUC
Click 4-gram	0.740
Click 4-gram (pretrained)	0.757
Click 4-gram (pretrained) + Video	0.784
Click 4-gram (pretrained) + Video (pretrained)	0.783

Results

We evaluate the performance of our proposed method by measuring AUC on the dropout prediction task. Each of four models listed in Table 1 uses different types of information and pretrained embeddings. The first model, Click 4-gram, does not use the video embeddings, but the click 4-gram embeddings in Figure 1. This is our state-of-the-art baseline for dropout prediction over weeks (Gardner and Brooks 2018). The second model, Click 4-gram (pretrained), also only uses the click 4-gram embeddings as the first model, but the embeddings are initialized with the pretrained embeddings shown in Figure 2a. The third model, Click 4-gram (pretrained) + Video, uses both the click 4-gram and video embeddings while only the click 4-gram embeddings are initialized with the pretrained embeddings in Figure 2a. The last model, Click 4-gram (pretrained) + Video (pretrained), uses both embeddings that are initialized with the pretrained embeddings in Figure 2a and Figure 2b.

We see a statistically significant increase in AUC ($p < 0.05$) from the first to the second model, which indicates pretraining click n-gram embeddings captures useful temporal relationships between click n-grams for dropout prediction. In addition, another statistically significant increase in AUC ($p < 0.05$) from the second to the third model tells that video embeddings capture meaningful correlation between videos and clicks. However, there is no improvement from the third to the fourth model, which concludes pretraining video embeddings does not help dropout prediction. We conjecture that pretraining video embedding barely learns the expressive video representation because it is too difficult to predict a video label from the clickstream. Or clickstream could be too noisy to provide meaningful information for predicting a video label. We leave designing a better objective for pretraining video representation as our future work.

References

Amnueypornsakul, B.; Bhat, S.; and Chinprutthiwong, P. 2014. Predicting attrition along the way: the uiuc model. In *EMNLP, Workshop on Analysis of Large Scale Social Interaction in MOOCs*.

Fei, M., and Yeung, D.-Y. 2015. Temporal models for predicting student dropout in massive open online courses. In *ICDMW*.

Gardner, J., and Brooks, C. 2018. Dropout model evaluation in moocs. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Glasmachers, T. 2017. Limits of end-to-end learning. *arXiv preprint arXiv:1704.08305*.

Halawa, S.; Greene, D.; and Mitchell, J. 2014. Dropout prediction in moocs using learner activity features. *Experiences and best practices in and around MOOCs*.

Kim, B.-H.; Vizitei, E.; and Ganapathi, V. 2018. Gritnet: Student performance prediction with deep learning. *arXiv preprint arXiv:1804.07405*.

Kloft, M.; Stiehler, F.; Zheng, Z.; and Pinkwart, N. 2014. Predicting mooc dropout over weeks using machine learning methods. In *EMNLP, Workshop on Analysis of Large Scale Social Interaction in MOOCs*.

Li, W.; Gao, M.; Li, H.; Xiong, Q.; Wen, J.; and Wu, Z. 2016. Dropout prediction in moocs using behavior features and multi-view semi-supervised learning. In *IJCNN*.

Lykourentzou, I.; Giannoukos, I.; Nikolopoulos, V.; Mparadis, G.; and Loumos, V. 2009. Dropout prediction in e-learning courses through the combination of machine learning techniques. *Computers & Education*.

Nagrecha, S.; Dillon, J. Z.; and Chawla, N. V. 2017. Mooc dropout prediction: lessons learned from making pipelines interpretable. In *Proceedings of the 26th International Conference on World Wide Web Companion*, 351–359. International World Wide Web Conferences Steering Committee.

Sinha, T.; Jermann, P.; Li, N.; and Dillenbourg, P. 2014a. Your click decides your fate: Leveraging clickstream patterns in MOOC videos to infer students’ information processing and attrition behavior. *CoRR* abs/1407.7131.

Sinha, T.; Li, N.; Jermann, P.; and Dillenbourg, P. 2014b. Capturing “attrition intensifying” structural traits from didactic interaction sequences of MOOC learners. In *EMNLP, Workshop on Analysis of Large Scale Social Interaction in MOOCs*.

Taylor, C.; Veeramachaneni, K.; and O’Reilly, U.-M. 2014. Likely to stop? predicting stopout in massive open online courses. *arXiv preprint arXiv:1408.3382*.

Wang, W.; Yu, H.; and Miao, C. 2017. Deep model for dropout prediction in moocs. In *Proceedings of the 2nd International Conference on Crowd Science and Engineering*, 26–32. ACM.

Whitehill, J.; Williams, J. J.; Lopez, G.; Coleman, C. A.; and Reich, J. 2015. Beyond prediction: First steps toward automatic intervention in mooc student stopout. *International Educational Data Mining Society*.

Whitehill, J.; Mohan, K.; Seaton, D.; Rosen, Y.; and Tingley, D. 2017. Delving deeper into mooc student dropout prediction. *arXiv preprint arXiv:1702.06404*.

Yang, D.; Sinha, T.; Adamson, D.; and Rosé, C. P. 2013. Turn on, tune in, drop out: Anticipating student dropouts in massive open online courses. In *NIPS, Data-driven education workshop*.

Yang, D.; Wen, M.; Howley, I. K.; Kraut, R. E.; and Rosé, C. P. 2015. Exploring the effect of confusion in discussion forums of massive open online courses. In *Learning @ Scale*.