Relaxed Clustered Hawkes Process for Procrastination Modeling in MOOCs

Mengfan Yao,[1] **Siqian Zhao,** [1] **Shaghayegh Sahebi,** [1] **Reza Feyzi Behnagh** [2]

[1] Department of Computer Science, University at Albany - SUNY
[2] Department of Educational Theory & Practice, University at Albany - SUNY
myao@albany.edu, szhao2@albany.edu, ssahebi@albany.edu,rfeyzibehnagh@albany.edu

Abstract

Hawkes processes have been shown to be efficient in modeling bursty sequences in a variety of applications, such as finance and social network activity analysis. Traditionally, these models parameterize each process independently and assume that the history of each point process can be fully observed. Such models could however be inefficient or even prohibited in certain real-world applications, such as in the field of education, where such assumptions are violated. Motivated by the problem of detecting and predicting student procrastination in students Massive Open Online Courses (MOOCs) with missing and partially observed data, in this work, we propose a novel personalized Hawkes process model (*RCHawkes-Gamma*) that discovers meaningful student behavior clusters by jointly learning all partially observed processes simultaneously, without relying on auxiliary features. Our experiments on both synthetic and real-world education datasets show that RCHawkes-Gamma can effectively recover student clusters and their temporal procrastination dynamics, resulting in better predictive performance of future student activities. Our further analyses of the learned parameters and their association with student delays show that the discovered student clusters unveil meaningful representations of various procrastination behaviors in students.

Introduction

Academic procrastination, or postponing the starting of planned studies, has been associated with negative side-effects on students' academic performance, psychological well-being, and health (Moon and Illingworth 2005; Steel 2007). This behavior is more prevalent in online educational settings, that require high levels of time-management and self-regulation skills (Lee and Choi 2011) and can lead to low academic outcomes and course drop-outs (Vitiello et al. 2018). With the growth of online education, it is essential to devise mechanisms to detect the potential future procrastination in students, to be able to prevent this behavior and its associated negative consequences.

In studies on self-reported academic procrastination, this behavior is indicated by cramming of studying activities: given a time interval followed by a deadline, students show limited studying activities at the beginning of the interval,

followed by a burst of studying (cramming) closer to the deadline Perrin et al. (2011); Gelman et al. (2016). However, these studies do not provide a unified quantitative definition of procrastination, other than qualitative student self-reports, that can be scarce and hard to obtain. Prior work also shows that although each student has their individual studying habits, students can be clustered into a few distinct groups by their studying behaviors (Yao, Sahebi, and Feyzi-Behnagh 2020; Uzir et al. 2020). In essence, in highly procrastinating students, getting closer to the deadline may trigger more intense studying activities, while in others, their studies are more regulated and distributed across the time interval. Despite these finding, most of the studies on student procrastination either ignored the temporal aspects of students' behavior (Cerezo et al. 2017; Kazerouni et al. 2017), or were not personalized for students (Baker, Evans, and Dee 2016; Park et al. 2018; Backhage, Ojeda, and Sifa 2017). More importantly, current research cannot predict when student's next activity will take place. Ideally, a procrastination model can capture the underlying *cluster structures* in student activity sequences, can be *personalized* to capture different students' studying habits, and can deal with *unseen data* such as assignments that are not yet started by students, and represent students' activity burstiness.

We note that Hawkes processes (Hawkes 1971) have the potential to represent students' procrastination behavior, as they model activity burstiness, as opposed to memoryless Poisson processes. However, when modeling one sequence per user-item pair, conventional Hawkes processes model each item's sequences individually and do not rely on the similarities between different items. Thus, they cannot infer parameters for items that have unseen data (Hosseini et al. 2016; Choi et al. 2015; Mei and Eisner 2017; Du et al. 2016). In some recent work, low-rank personalized Hawkes models aim to address this problem (Du et al. 2015b), usually with the help of auxiliary features to reinforce the low-rank assumption (Shang and Sun 2018, 2019). Yet, to the best of our knowledge, none of the previous Hawkes models were able to represent the cluster structure between sequences, while being personalized and inferring unseen data.

In this paper, we propose a novel Relaxed Clustered Hawkes process with a Gamma prior (*RCHawkes-Gamma*) to model and predict the cramming procrastination behavior in students of Massive Open Online Courses (MOOCs). To

do this, we model each student-assignment pair, that is the interactions of a student with a course assignment characterized by activity times, as a uni-variate Hawkes process. By modeling all student-assignment pairs jointly, our proposed model is able to capture similarities shared among students (i.e. cluster structures) by learning a low-dimensional representation of procrastination (i.e. personalization). As a result, even for student-assignment pairs without observed history (i.e. unseen data), their parameters can be inferred based on the group structure, without relying on auxiliary features or historical observations.

It is worth noting that our proposed approach can be useful in real-world scenarios such as for professional educators or adaptive learning systems. As an example, the prediction of future activities especially on the unseen student-assignment pairs can provide teachers the opportunity to intervene with students who show strong procrastination tendencies. For students, their learning activities can be presented in formats such as a dashboard, for visualization, summarization, and feedback generation, which in turn can be beneficial in regularizing students' learning behaviors. While our model is created with the education domain in mind, it can be applied to other domains, such as recommender systems.

To summarize, our contributions are: (1) We propose a Relaxed Clustered Hawkes model, driven by the problem of modeling academic procrastination in MOOCs; (2) Our *personalized* model represents the *similarity structure* between multiple event sequences without requiring auxiliary features (Section) and infers *unseen data* in event sequences; (3) We experiment on both synthetic and real-world datasets to show that the proposed model can recover clusters of students and their temporal procrastination dynamics, resulting in a better predictive performance of future activities (Section); and (4) We further study the learned parameters to demonstrate that the discovered student clusters are meaningful representations of various procrastination-related behaviors in students (Section).

Related Work

Low-Rank Hawkes Processes Hawkes processes (Hawkes 1971) have been successfully used in applications such as social networks (e.g. Chen et al. 2019), mobility patterns (e.g. Vassøy et al. 2019), and recommender systems (e.g. Du et al. 2015b). Among them, the most relevant work to ours is low-rank uni-variate Hawkes processes proposed by Du et al., to capture user-item interaction and recommend items "at the right moment" (Du et al. 2015b). However, this work does not incorporate the clustering behavior that is essential in our problem domain.

Other relevant literature on Hawkes processes mainly falls into 3 categories: (1) Multi-variate Hawkes processes that focus on modeling the mutual excitation among sequences (Zhou, Zha, and Song 2013; Luo et al. 2015; Bacry et al. 2020; Lemonnier, Scaman, and Kalogeratos 2017). (2) Uni-variate Hawkes models that model each sequence *independently* and discard the potential relatedness among all sequences, thus cannot infer sequence's future when its history is not observed e.g. (Mei and Eisner 2017; Du et al.

2016, 2015a; Xiao et al. 2017; Li, Wei, and Ke 2018; Li and Ke 2020). For example, Du et al. propose to use RNN to model the arrival times of a given sequence to capture more complicated sequence dynamics compared to traditional Hawkes models (Du et al. 2016). Such RNN-based models predict future time after time t based on the observed history unfolded up to time t, therefore cannot directly predict sequences that do not have historical observations; (3) Approaches that *jointly* model different sequences as uni-variate Hawkes processes by capturing the similarities among the sequences (e.g. via a low-rank constraint). Therefore, they can predict the future events for the sequences without historical observations, by utilizing histories from sequences that are structurally similar. However, such methods usually rely on auxiliary information (He et al. 2015; Li, Wei, and Ke 2018; Shang and Sun 2018, 2019). For example, in the recommender system setting, Shang et al. impose a local low-rank constraint on the parameter matrix to model large-scale user-item interactions by first computing user-item pairs' similarities via item features (Shang and Sun 2018). In contrast, due to privacy constraints in our application, many educational datasets are highly anonymized and scarce. Consequently, having a model that does not require such information is valuable in our context.

Procrastination Modeling in Education Domain As there is no quantitative definition for procrastination behavior, in most of the recent educational data mining literature, procrastination-related behavior has been summarized by curating time-related features from student interactions in the course. These studies aim to evaluate the relationships between these time-related features with student performance and do not model temporal aspects of procrastination (Baker, Evans, and Dee 2016; Cerezo et al. 2017; Kazerouni et al. 2017; Agnihotri, Baker, and Stalzer 2020). The few recent works that model student activity sequences, are limited in their assumptions, do not capture student activity intensities, are not personalized, do not model time dependencies between student actions, and do not infer missing data (Park et al. 2018; Yao, Sahebi, and Feyzi-Behnagh 2020). For example, Park et al. classify students to procrastinators vs. non-procrastinators by formulating a measure using a mixture model of per-day student activity counts during each week of the course (Park et al. 2018). But, it cannot model non-homogeneously spaced deadlines in a course. Furthermore, even though each student's activity is counted in a daily basis, it is not a temporal approach that models activity time points. Indeed, none of these models can predict *when* the future activities will happen. Similarly, Backhage et al. proposed Poisson distribution to model students' daily activity count in order to capture procrastination-deadline cycles of all students in the course (Backhage, Ojeda, and Sifa 2017). In their work, each day of the week is associated with a Poisson rate parameter that is constant during the whole course. Despite representing individual student activity counts, this model cannot differentiate between different weeks in the course, does not have a continuous time scale, and cannot capture non-homogeneously spaced deadlines in a course.

To the best of our knowledge, the only model that can be

compared to ours in predicting activity times is a Hawkes process model by Yao et al. (Yao, Sahebi, and Feyzi-Behnagh 2020) that relates procrastination to the mutual excitation among activity types. This work does not model student behavior clusters, and cannot infer unseen data. We use this model, called EdMPH, as one of our baselines.

Problem Formulation

Our goal is to model partially observed student-assignment interactions and predict two types of future student-assignment interactions: 1) future assignments with no historical activities (*unseen data requirement*), and 2) current assignments that students are working on (assignment-student pairs with *partially observed* history).

Specifically, we consider a course that includes N non-parallel assignments and M students. Each student u_j can perform a sequence of activities towards each assignment a_i, such that each sequence is indexed by a student-assignment pair (a_i, u_j). Activities in a sequence are presented with a timestamp that marks their arrival time. We assume that the activities within each student-assignment pair happen either because they are a pre-requisite for another assignment-related activity (*internal stimuli*), or because of a non-activity related reason (*external stimuli*). For an example of internal stimuli, think of when students divide their big tasks (e.g., submitting the final assignment response) into smaller sub-tasks (e.g., solving a sub-problem of the assignment), within each sub-task, one activity spontaneously leads to another related activity. Conversely, external stimuli can come from the individual student's tendency to study regularly or due to the assignment deadline [1]. On the other hand, we assume no causal relationship between student-assignment pairs: since assignments are not parallel, activities towards assignments do not trigger each other. Further, since students do not work in teams and are not in a social setting, there are no triggering effects between student activities. We also assume that while students having their individual learning pattern towards each assignment (*personalization assumption*), their studying activities follow a latent structure that can group students with similar learning behaviors (*cluster assumption*).

Model: Relaxed Clustered Hawkes

According to our problem formulation and assumptions, we build our model based on uni-variate Hawkes processes. The reason behind our choice of the model is two-fold: (1) Unlike the memoryless Poisson process that assumes the independence among activities, Hawkes can model the aforementioned internal and external stimuli that exist in student activities; (2) Unlike the multi-variate Hawkes processes that assume triggering effects between dimensions, there are no exciting effects between assignments or student sequences. We first present the intensity function that defines student-assignment pairs. We then add low-rank and relaxed clustering constraints to capture our personalization

---

[1]As student activities are triggered by the upcoming deadlines from the future but not the past, without loss of generalizability, we use a reversed activity timeline for our data.

and cluster assumptions, and add a Gamma prior to address the unseen data requirement.

Uni-Variate Hawkes Intensity Function

Formally, given a sequence of activities for an assignment-student pair (a_i, u_j), we model its activities' arrival times $X_i^j = \{x_{i\tau}^j | \tau = 1, ..., n_i^j\}$ by a uni-variate Hawkes process, via the intensity function of time t, defined as follows (Hawkes 1971):

$$\lambda(t)_{ij} = U_{ij} + A_{ij}\beta \sum_{\tau=1}^{n_{ij}} \exp(-\beta(t - x_{i,\tau}^j)), \quad (1)$$

where $x_{i,\tau}^j$ is the τ-th element in the vector $X_i^j \in \mathbb{R}^{n_{ij}}$, which denotes the arrival time of the τ-th activity that belongs to assignment-student pair (a_i, u_j), n_{ij} is the total number of observed activities for (a_i, u_j); $U \in \mathbb{R}^{N \times M}$ is the non-negative base rate matrix, where U_{ij} quantifies the expected number of activities that are triggered externally within (a_i, u_j); $A \in \mathbb{R}^{N \times M}$ is the non-negative self-excitement matrix, with A_{ij} representing the self-exciting or bursty nature of (a_i, u_j), i.e., the expected number of activities that are triggered by the past activities; and β is a global decay rate that represents how fast the historical activities stop affecting the future activities.

Relaxed Clustered Hawkes

Conventional uni-variate Hawkes processes model each process individually. In this work, we assume that the occurrences of assignment activities and their characteristics, parameterized by Hawkes process, are similar among some students, but less similar to some others, i.e. parameter matrix A exhibits cluster structure on its columns.

Particularly, we assume that students form $k < M$ clusters according to their behaviors towards all assignments represented in A's column vectors. To impose this, we add a clustering constraint to our model using the sum of squared error (SSE) penalty, similar to K-means clustering:

$$P(A, W) = \rho_1 \text{tr}(A^\top A - W^\top A^\top AW) + \rho_2 \text{tr}(A^\top A) \quad (2)$$
$$= \text{tr}(A((1 + \frac{\rho_1}{\rho_2})I - WW^T)A^\top,$$

where ρ_1 and ρ_2 are regularization coefficients; $W \in \mathbb{R}^{M \times k}$ is an orthogonal cluster indicator matrix, with $W_{ij} = \frac{1}{\sqrt{n_j}}$ if i is in j-th cluster, and 0 otherwise (showing which students belong to which cluster); and n_j is the size of cluster j.

Since this strict constraint is non-convex, we follow Jacob et al.'s work (Jacob, Vert, and Bach 2009) to obtain its convex relaxation problem:

$$\min \mathcal{L}_c(A, Z) = \min \frac{\rho_2(\rho_2 + \rho_1)}{\rho_1} tr(A(\frac{\rho_1}{\rho_2}I + Z)^{-1}A^\top) \quad (3)$$
$$\text{s.t. } tr(Z) = k, Z \preceq I, Z \in S_+^M.$$

$Z = WW^\top \in \mathbb{R}^{M \times M}$ represents cluster-based similarity of students, with W defined in Eq. 2. Here, the trace norm is a surrogate of the original assumption that there are k clusters and the other two constraints are the relaxation of W being orthogonal. As a result, this equation is jointly convex to both A and Z. We call this model *RCHawkes*.

A mixture Gamma prior

To improve our model's robustness to potential outliers and to possibly reduce overfitting, we add a mixture Gamma prior on the self-excitement matrix A. As a result, the summation of the first three terms in Equation 8 is the A-Posteriori estimation, which not only is more robust comparing with Maximum Likelihood Estimation, also it provides an interpretation of each component's hyperparameters in student clusters: i.e. the pseudo counts of externally and internally excited activities. Specifically, consider the prior for A_{ij} when student i is in m-th cluster:

$$p(A_{ij}; \Theta_m) = \frac{1}{\Gamma(s_m)\theta_m^{s_m}} A_{ij}^{s_m-1} \exp(-\frac{A_{ij}}{\theta_m^{s_m}}), \quad (4)$$

where $\Theta_m = (s_m, k_m)$, are hyperparameters which respectively control the shape and the scale of the gamma distribution in cluster m. The loss brought by the mixture Gamma prior can be computed as follows:

$$\mathcal{L}_g = \log p(A; \Theta_1, .., \Theta_k) \quad (5)$$

$$= \sum_{X_i^j \in \mathcal{O}} \Big[\log \sum_{m=1}^{k} \frac{1}{k} \frac{1}{\Gamma(s_m)\theta_m^{s_m}} A_{ij}^{s_m-1} \exp(-\frac{A_{ij}}{\theta_m^{k_m}}) \Big],$$

where $\mathcal{O}$ is the collection of all observed X_i^j.

Objective Function

For our model, we need to consider the multiple sequences (as in Eq. 1) and add the introduced constraints. Here we first introduce a recursive function R that can be computed offline to ease the computation. It is set to be 0 when τ is 1 and updated recursively via the following equation.

$$R_{ij}(\tau) = \big(1 + R_{ij}(\tau-1)\big) \exp(-\beta(x_{i,\tau}^j - x_{i,\tau-1}^j)) \quad (6)$$

We also construct the matrix T as follows to avoid repetitive computation in iterations:

$$T = [\sum_{\tau=1}^{n_{ij}} (\exp(-\beta(x_{i,n_{ij}}^j - x_{i,\tau}^j)) - 1)]_{N \times M} \quad (7)$$

To this end, the final objective function of our proposed model, given the observed activities for all assignment-student pairs X can be described as in Eq. 8.

$$\min_{A \geq 0, U \geq 0, Z} -L(X; A, U) \quad (8)$$

$$= - \sum_{X_i^j \in \mathcal{O}} \sum_{\tau=1}^{n_{ij}} \log \big(U_{ij} + A_{ij}\beta R_i^j(\tau)\big) + U_{ij}x_{i,n_{ij}}^j$$

$$+ A \circ T - \mathcal{L}_g(A; \Theta_1, .., \Theta_k) + \mathcal{L}_c(A, Z) + \rho_3 tr(A)$$
$$\text{s.t. } A \geq 0, U \geq 0,$$

where $\mathcal{L}_c$ and $\mathcal{L}_g$ are the previously defined losses introduced by clustering and gamma prior respectively and ρ_3 is a regularization coefficient.

The trace norm regularization, is a convex surrogate for computing rank of A, which enables the knowledge transfer from the processes with observations to the unseen assignment-user pairs that do not have any observed historical activities. Finally, to not violate the definition of Hawkes process, we have non-negative constraints on A and U.

Optimization

To solve the minimization problem in Eq. 8, we could use the Stochastic Gradient Descent algorithms. However, the non-negative constraints on A and U along with the non-smoothed trace norms can complicate the optimization. To tackle this problem, we used the Accelerated Gradient Method (Nesterov 2013). The key component of using this method is to compute the following proximal operator:

$$\min_{A_z, U_z, Z_z} \|A_z - A_s\|_F^2 + \|U_z - U_s\|_F^2 + \|Z_z - Z_s\|_F^2 + L \quad (9)$$

$$\text{s.t. } tr(Z_z) = k, tr(A_z) \leq c, A_z \geq 0, U_z \geq 0, Z_z \preceq I, Z_z \in S_+^M$$

where subscripts z and s respectively represents the corresponding parameter value at the current iteration and search point (Nesterov 2013). To efficiently solve the objective function, we use the Accelerated Gradient Descent framework. [2]

Experiments

In this section we evaluate our approach with several state-of-the-art competitors on both simulated and real datasets.

Setup. In simulated data, we randomly select a ratio of $r = [0.1, 0.3, 0.5, 0.7]$ amount of students' last two assignment activities to be entirely missing (unseen set), and for the rest of the student-assignment pairs, the first 70% of activities are used for training (training set) and the last 30% are used for testing (seen set). In both real datasets, the unit time is set to be an hour, and we use activities that took place before the mid point of the last assignment as training. Hyperparameters of the proposed and baseline models are tuned via grid search.

Baselines

We consider two sets of state-of-the-art baselines: the ones that are able to infer unseen data, and the ones that cannot. A summary of all baseline approaches is presented in Table 1. In the following we briefly introduce each of the baselines.

EdMPH: A Hawkes model that was recently proposed to model student procrastination in Educational Data Mining domain (Yao, Sahebi, and Feyzi-Behnagh 2020). It applies a Multivariate Hawkes Model which utilizes student activity types as extra information, and cannot infer unseen data.

RMTPP: A Recurrent Neural Network Hawkes model to represent user-item interactions (Du et al. 2016). It does not directly infer parameters of unseen data and it uses activity markers (i.e. features) as an input.

ERPP: A similar approach to baseline RMTPP, but it includes time-series loss in the loss function (Xiao et al. 2017).

HRPF and **DHPR**: Two Poisson factorization models proposed in (Hosseini et al. 2018) that do not require user-network as auxiliary features. These models, however, do not directly model the time-dependencies between the future and the past, thus cannot quantify activity self-excitement.

HPLR: An item recommendation model using Hawkes process (Du et al. 2015b). It is the most similar to ours, as it

---

[2]Details of the algorithm, its complexity, convergence analyses, and our code can be found in https://github.com/persai-lab/AAAI2020-RCHawkes-Gamma.

Application	Model	Infer Unseen Data	Require No External Features	Model Time Dependency
EDM	RCHawkes-Gamma	✓	✓	✓
	RCHakwes	✓	✓	✓
	EdMHP	✗	✗	✓
Rec-Sys	HPLR	✓	✓	✓
	ERPP	✗	✗	✓
	HRPF	✓	✓	✗
	DRPF	✓	✓	✗
	RMTPP	✗	✗	✓

Table 1: A summary of baseline approaches.

imposes a low rank assumption on matrices A and U and can infer unseen data. However, unlike our model, it does not consider the cluster structure of parameter matrix A.

RCHawkes: A variation of our proposed model that does not use a Gamma prior. Its objective is to find the maximum of log-likelihood rather than the maximum of A-posterior.

Datasets

Synthetic Data To create simulated student-assignment pairs, we first construct the parameter matrices. We build A_s by: a) sampling $k = 3$ sets of column α's from different Gamma distributions, for different student procrastination behavior clusters; b) adding white noise ($\sigma^2 = 0.1$); and c) shuffling all columns randomly to break the order. We build U_s, by sampling it from a normal distribution. Then, we sample 150 activities for each assignment-student pair using the Ogata thinning algorithm (Ogata 1988), which is the most commonly used sampling method in the related literature. Finally, we obtain 5400 simulated student-assignment pairs and 810K synthetic activities.

Computer Science Course on Canvas Network (CANVAS) This real-world MOOC dataset is from the Canvas Network platform (Canvas-Network 2016). Canvas Network is an online platform that hosts various open courses in different academic disciplines. The computer science course we use happens during $\sim$ 6 weeks. In each week, an assignment-style quiz is published in the course resulting in 6 course assignments. In total, we extract $\sim$ 740K assignment-related activity timestamps from 471 students.

Big Data in Education on Coursera (MORF) Our second real-world dataset is collected from an 8-week "Big Data in Education" course in Coursera platform. The dataset is available through the MOOC Replication Framework (MORF) (Andres et al. 2016). In total, we extract $\sim$ 102K activities of 675 students related to 8 assignments.

Fit and Prediction Performance of RCHawkes-Gamma

Estimated Parameters on Simulated Data In the simulated dataset, as we know the true parameters (i.e. A and U), we compare the Root Mean Squared Error (RMSE) of estimated $\hat{A}$ and $\hat{U}$, varying unseen data ratio r [3]. The results are presented in Tbl. 2. RCHawkes-Gamma and RCHakwes outperform the baseline methods usually by a large margin, for both the sequences with seen and unseen history. Also,

---

[3]Baselines ERPP, RTMPP, and EdHawkes cannot be used in this analysis, since they parameterize the processes differently.

even though all models perform worse with the increase of r, RCHawkes-Gamma and RCHakwes' RMSEs have a lower standard deviation, indicating less variation in their performance even in high missing data ratios. Additionally, the models' performances in unseen data are generally worse than their performances in the processes with observed historical activities. One possible reason is that the Hawkes parameters for unseen data in this simulation can only be inferred from the similar processes with observed data by leveraging the row and column relatedness, while the true characteristics of the unseen processes can not be entirely captured as there are no observations that the models can use.

Clustering Structure of Hawkes Parameters To see if the cluster structure of students is well captured by each model, we compute and present the correlation matrix of $\hat{A}$ between students with the recovered cluster orders in Figure 3. Our proposed models recover this structure closer to the ground truth (Figure 3 (a)), i.e. a higher correlation within clusters (darker blocks) and a lower correlation between clusters (lighter blocks). HPLR introduces unnecessary correlations between clusters, possibly because of not having the student cluster assumption. HRPF simply assumes all assignment-student pairs share the same parameter, thus has a meaningless correlation of 1 among all students. Finally, although DRPF improves HRPF by considering activity self-excitements, it fails to capture any meaningful correlation within clusters.

Returning Time Prediction on Simulated and Real Data In these experiments, we use a popular metric that has been used in many Hawkes-based models, i.e. RMSE on Time Prediction (TP), where TP is defined on the estimated next activity time, given the observed history (e.g. (Du et al. 2013)). The baselines that do not directly infer parameters on unseen data (the future assignment scenario) are not included in the unseen data evaluation. Following the method used in (Du et al. 2013), we sampled future activities based on the learned parameters via Ogata's thinning algorithm.

Figures 1 and 2 respectively present the prediction error and 95% confidence interval on simulated and real-world data. As we can see from Figure 1, the proposed methods RCHawkes-Gamma and RCHakwes consistently outperform other baselines in all settings, except when the missing ratio is 0.1. In that case, RMTPP and ERPP achieve the smallest error in seen data. However, unlike the proposed models that are almost invariant to the increase of missing ratio, ERPP and RMTPP's performances change dramatically with increasing r. More importantly, they lack the ability to directly predict the next activity time when the activity history is unseen. When comparing the baselines in the real datasets (Figure 2), all approaches perform better in CANVAS dataset, compared to MORF. One possible explanation is that, in CANVAS, each assignment-student pair contains more historical activity time stamps. Therefore it provides all approaches more training data than the MORF dataset.

Student Procrastination in RCHawkes-Gamma

In the following section, we will focus to finding the connections between the characteristics of students' learning activ-

	Model	r = 0.1		r = 0.3		r = 0.5		r = 0.7	
		seen	unseen	seen	unseen	seen	unseen	seen	unseen
RMSE for A	RCHawkes-Gamma	**0.094±0.024**	**0.102±0.037**	0.121±0.017	**0.114±0.056**	0.141±0.033	0.139±0.033	**0.136±0.077**	**0.137±0.052**
	RCHawkes	0.108±0.017	0.108±0.054	**0.115±0.024**	0.116±0.039	**0.126±0.033**	**0.136±0.033**	0.180±0.072	0.170±0.048
	HPLR	0.631±0.110	0.663±0.331	0.645±0.141	0.607±0.216	0.635±0.133	0.633±0.133	0.634±0.304	0.634±0.204
	HRPF	0.664±0.769	0.664±0.769	0.664±0.770	0.664±0.770	0.663±0.769	0.663±0.770	0.664±0.769	0.664±0.767
	DRPF	0.474±0.461	0.474±0.461	0.479±0.465	0.479±0.465	0.473±0.462	0.473±0.462	0.474±0.463	0.474±0.463
RMSE for U	RCHawkes-Gamma	0.075±0.022	0.085±0.036	**0.069±0.017**	**0.060±0.050**	**0.062±0.030**	**0.064±0.030**	0.071±0.039	0.075±0.026
	RCHawkes	0.074±0.020	0.089±0.061	0.074±0.020	0.075±0.032	0.077±0.030	0.079±0.030	**0.069±0.026**	**0.062±0.017**
	HPLR	0.110±0.082	**0.078±0.047**	0.081±0.060	0.078±0.094	0.091±0.035	0.091±0.035	0.090±0.096	0.095±0.065
	HRPF	0.105±0.055	0.311±0.055	0.119±0.068	0.183±0.068	0.141±0.071	0.142±0.071	0.179±0.068	0.120±0.070
	DRPF	**0.062±0.052**	0.300±0.035	0.088±0.049	0.165±0.045	0.121±0.051	0.121±0.050	0.167±0.053	0.102±0.054

Table 2: RMSE (±error standard deviation) of $\hat{A}$ and $\hat{U}$ on seen and unseen data, with various missing data ratios (r)
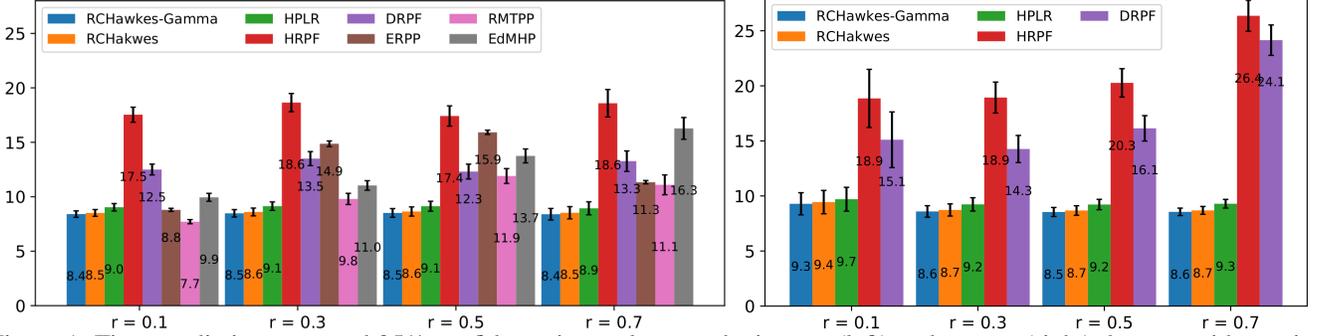

Figure 1: Time prediction error and 95% confidence interval on synthetic seen (left) and unseen (right) datasets with varying data missing ratios (r)
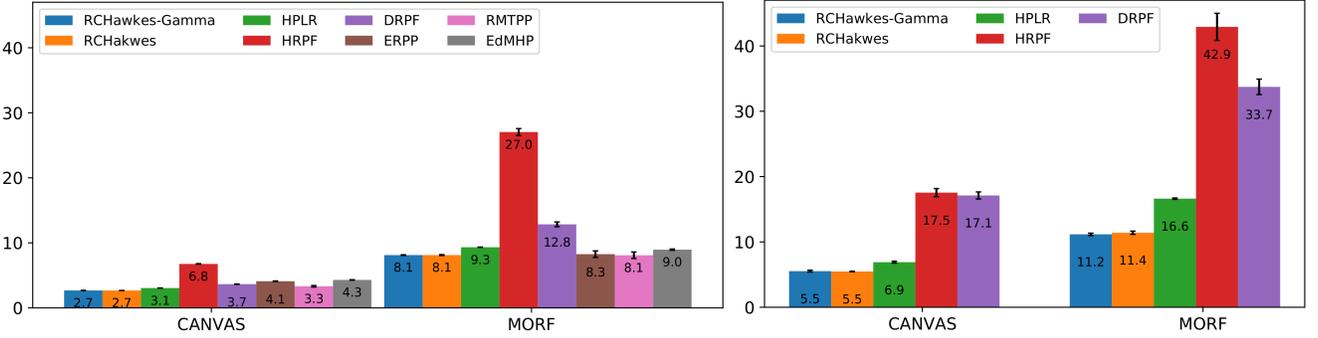

Figure 2: Time prediction error on seen (left) and unseen (right) data with 95% confidence interval on real-world datasets

ities (parameterized by our model) and students' cramming behaviors.

Students' Cramming Behaviors Since procrastination does not have a quantitative definition, in the first step of our analysis, we define the following measure to describe the degree of student procrastination presented in MOOCs: delay $= \frac{t_{ij}^a - t_{ij}^s}{t_{ij}^d - t_{ij}^s}$ to quantify student j's normalized delay in starting any activity that is associated with assignment i, where superscript s, a, d respectively represents the start of the assignment, the first, and the last activity in the student-assignment pair. Intuitively, this measure is the absolute time that student j delays in starting assignment i, normalized by the duration that assignment i is available for student j. Note that this measure is just a simple representation and cannot replace our model in predicting next activity times or uncovering cluster structures.

Correlation Analysis In order to show how students activities parameterized by Hawkes and student delays are associated, we compute the Spearman's rank correlation co-efficient between each pair of the variables. We choose the Spearman's correlation because it does not assume a normal distribution on the parameters, nor a linear relationship between the variables as Pearson correlation does. As we can see in Table 3, the two-sided p-values suggest that the correlations between these variables are statistically significant. We can also see that all the correlation coefficients are positive, meaning that student delays are positively associated with the base rate, i.e. expected number of occurrences per unit time that are excited by external stimuli (for example deadlines), and the burstiness of the occurrences. On the other hand, by looking at the two courses side-by-side, we can see that delay is more strongly associated with α in CANVAS. But, its association with the base rate μ is stronger in MORF. This suggests two different kinds of relationships between students and assignments: while in CANVAS big bursts of activities might suggest delays, in MORF small but frequent activities are associated with student delays.
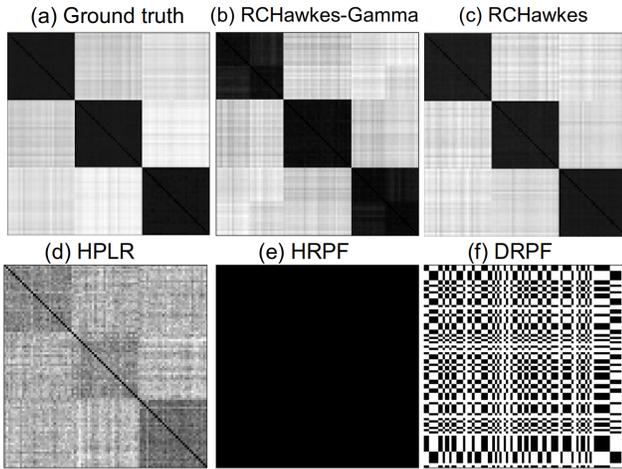
Figure 3: The ground truth of A's correlation matrix (a), and the estimated $\hat{A}$'s correlation matrix learned by each model.

		α	μ	delay
CANVAS	α	1		
	μ	0.284***	1	
	delay	0.345***	0.144***	1
MORF	α	1		
	μ	0.243***	1	
	delay	0.264***	0.412***	1

Table 3: Spearman's correlation between learned parameters and computed normalized student delays. p<0.001*** p<0.01 ** p<0.05*

Clustering Analysis To see if there are meaningful differences in the delay measure for students in different clusters, we first cluster the students using the K-means clustering algorithm, which has a similar objective to the cluster-structure term in our model (Eq. 2), on the learned $\hat{A}$ matrix. Specifically, student u_j is represented by the vector of estimated self-excitement parameters $(\hat{\alpha}_{1,i}, ..., \hat{\alpha}_{N,i})$ that are learned by RCHawkes-Gamma, and the cluster number for K-means is decided via grid search by looking at SSE. To examine the possible differences between clusters of students in terms of student delays, we conduct the Kruskal-Wallis test on all student delays across the clusters for each assignment. We report the average delay of all students in each cluster and for each assignment. The results are shown in Table 4 for CANVAS and in Table 5 for MORF dataset. In CANVAS, 4 student clusters are found. These clusters all have significant differences in terms of delays. For example, students in cluster 1 have the smallest delay, with a

Assign. #.	cluster 1	cluster 2	cluster 3	cluster 4	p-value
size	81	144	207	39	-
1	0.3335	0.4583	0.6108	0.9064	1.34E-16***
2	0.6245	0.5788	0.8476	1.0854	3.59E-09***
3	0.6911	0.7143	0.8633	0.9655	4.36E-05***
4	0.6050	0.6958	0.8515	1.0717	0.0008***
5	0.5969	0.7080	0.9084	1.1217	0.0195*
6	0.5351	0.7647	0.9002	1.0970	0.0149*

Table 4: Kruskal Wallis test on delays in different clusters in CANVAS dataset. p<0.001*** p<0.01 ** p<0.05*

Assign. #.	cluster 1	cluster 2	cluster 3	p-value
size	573	34	68	-
1	0.4991	0.6710	0.4477	2.30E-09***
2	0.5120	0.7288	0.4855	1.90E-08***
3	0.5570	0.6904	0.6105	7.50E-05***
4	0.4699	0.6122	0.5360	0.0004***
5	0.5626	0.6358	0.6308	0.0070***
6	0.5329	0.6236	0.6642	8.56E-06***
7	0.4325	0.5598	0.7672	2.12E-20***
8	0.3974	0.5172	0.7629	3.84E-27***

Table 5: Kruskal Wallis test on delays in different clusters in MORF dataset. p<0.001*** p<0.01 ** p<0.05*

general decreasing trend towards the later assignments. On the other hand, delays are the worst for students in cluster 4, with an average delay greater or close to 1 for all assignments, which implies that this group of students tend to start the assignments very close to or even later than the deadline. In the 3 clusters that are found in the MORF dataset, the p-values of Kruskal-Wallis tests show strong evidence of cluster differences for each assignment. Specifically, the majority of the students in the MORF course are in cluster 1 and their delays are overall the lowest comparing to the other two clusters. They tend to delay less and less over time. On the other hand, students in cluster 3 start the course with a low delay but increase their delay so fast that at the end of the course, they turn out to be the students who delay the most. This analysis demonstrates that the self-excitement parameters have strong associations with student delays, which not only reinforces the findings from the correlation analysis, but also suggests that they are good indicators in characterizing students' cramming behaviors.

Conclusion

In this paper, we proposed a novel uni-variate clustered Hawkes process model, RCHawkes-Gamma to model procrastination behaviors of students in MOOCs assuming a cluster structures between students. We test our proposed model on a synthetic dataset and two real-world MOOC datasets. The results of our experiments show that our proposed model can predict students' next activity time with lower time prediction error on both seen and unseen data, compared to the baseline methods. We also study and analyze the parameters learned by the proposed model on both MOOC datasets. Our analysis reveals the positive associations between student delays with our model's parameters. The model also discovers meaningful clusters of students who show different delaying behavior trends. A limitation of this work is that the delay measure is used as a proxy for procrastination, while self-reported procrastination measures could have helped in labeling delays more accurately as procrastination.

Acknowledgements

References

Agnihotri, L.; Baker, R. S.; and Stalzer, S. 2020. A Procrastination Index for Online Learning Based on Assignment Start Time. In *The 13th International Conference on Educational Data Mining*.

Andres, J. M. L.; Baker, R. S.; Siemens, G.; Gašević, D.; and Spann, C. A. 2016. Replicating 21 findings on student success in online learning. *Technology, Instruction, Cognition, and Learning* 313–333.

Backhage, C.; Ojeda, C.; and Sifa, R. 2017. Circadian Cycles and Work Under Pressure: A Stochastic Process Model for E-learning Population Dynamics. In *Data Science–Analytics and Applications*, 13–18. Springer.

Bacry, E.; Bompaire, M.; Gaïffas, S.; and Muzy, J.-F. 2020. Sparse and low-rank multivariate Hawkes processes. *Journal of Machine Learning Research* 21(50): 1–32.

Baker, R.; Evans, B.; and Dee, T. 2016. A Randomized Experiment Testing the Efficacy of a Scheduling Nudge in a Massive Open Online Course (MOOC). *AERA Open* 2(4). ISSN 2332-8584, 2332-8584.

Canvas-Network. 2016. Canvas Network Courses, Activities, and Users (4/2014 - 9/2015) Restricted Dataset. doi: 10.7910/DVN/XB2TLU.

Cerezo, R.; Esteban, M.; Sánchez-Santillán, M.; and Núñez, J. C. 2017. Procrastinating Behavior in Computer-Based Learning Environments to Predict Performance: A Case Study in Moodle. *Frontiers in Psychology* 8. ISSN 1664-1078.

Chen, H.; Liu, R.; Park, N.; and Subrahmanian, V. 2019. Using twitter to predict when vulnerabilities will be exploited. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 3143–3152. ACM.

Choi, E.; Du, N.; Chen, R.; Song, L.; and Sun, J. 2015. Constructing disease network and temporal progression model via context-sensitive hawkes process. In *2015 IEEE International Conference on Data Mining*, 721–726. IEEE.

Du, N.; Dai, H.; Trivedi, R.; Upadhyay, U.; Gomez-Rodriguez, M.; and Song, L. 2016. Recurrent marked temporal point processes: Embedding event history to vector. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1555–1564.

Du, N.; Farajtabar, M.; Ahmed, A.; Smola, A. J.; and Song, L. 2015a. Dirichlet-hawkes processes with applications to clustering continuous-time document streams. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 219–228.

Du, N.; Song, L.; Woo, H.; and Zha, H. 2013. Uncover topic-sensitive information diffusion networks. In *Artificial Intelligence and Statistics*, 229–237.

Du, N.; Wang, Y.; He, N.; Sun, J.; and Song, L. 2015b. Time-sensitive recommendation from recurrent user activities. In *Advances in Neural Information Processing Systems*, 3492–3500.

Gelman, B.; Revelle, M.; Domeniconi, C.; Johri, A.; and Veeramachaneni, K. 2016. Acting the Same Differently: A Cross-Course Comparison of User Behavior in MOOCs. *International Educational Data Mining Society* .

Hawkes, A. G. 1971. Spectra of some self-exciting and mutually exciting point processes. *Biometrika* 58(1): 83–90.

He, X.; Rekatsinas, T.; Foulds, J.; Getoor, L.; and Liu, Y. 2015. Hawkestopic: A joint model for network inference and topic modeling from text-based cascades. In *International conference on machine learning*, 871–880.

Hosseini, S. A.; Khodadadi, A.; Alizadeh, K.; Arabzadeh, A.; Farajtabar, M.; Zha, H.; and Rabiee, H. R. 2018. Recurrent poisson factorization for temporal recommendation. *IEEE Transactions on Knowledge and Data Engineering* 32(1): 121–134.

Hosseini, S. A.; Khodadadi, A.; Arabzadeh, A.; and Rabiee, H. R. 2016. Hnp3: A hierarchical nonparametric point process for modeling content diffusion over social media. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, 943–948. IEEE.

Jacob, L.; Vert, J.-p.; and Bach, F. R. 2009. Clustered multi-task learning: A convex formulation. In *Advances in neural information processing systems*, 745–752.

Kazerouni, A. M.; Edwards, S. H.; Hall, T. S.; and Shaffer, C. A. 2017. DevEventTracker: Tracking Development Events to Assess Incremental Development and Procrastination. In *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education - ITiCSE '17*, 104–109. Bologna, Italy: ACM Press.

Lee, Y.; and Choi, J. 2011. A review of online course dropout research: Implications for practice and future research. *Educational Technology Research and Development* 59(5): 593–618.

Lemonnier, R.; Scaman, K.; and Kalogeratos, A. 2017. Multivariate Hawkes processes for large-scale inference. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Li, T.; and Ke, Y. 2020. Tweedie-Hawkes Processes: Interpreting the Phenomena of Outbreaks. In *AAAI*, 4699–4706.

Li, T.; Wei, P.; and Ke, Y. 2018. Transfer hawkes processes with content information. In *2018 IEEE International Conference on Data Mining (ICDM)*, 1116–1121. IEEE.

Luo, D.; Xu, H.; Zhen, Y.; Ning, X.; Zha, H.; Yang, X.; and Zhang, W. 2015. Multi-task multi-dimensional hawkes processes for modeling event sequences. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Mei, H.; and Eisner, J. M. 2017. The neural hawkes process: A neurally self-modulating multivariate point process. In *Advances in Neural Information Processing Systems*, 6754–6764.

Moon, S. M.; and Illingworth, A. J. 2005. Exploring the dynamic nature of procrastination: A latent growth curve analysis of academic procrastination. *Personality and Individual Differences* 38(2): 297–309.

Nesterov, Y. 2013. Gradient methods for minimizing composite functions. *Mathematical Programming* 140(1): 125–161.

Ogata, Y. 1988. Statistical models for earthquake occurrences and residual analysis for point processes. *Journal of the American Statistical association* 83(401): 9–27.

Park, J.; Yu, R.; Rodriguez, F.; Baker, R.; Smyth, P.; and Warschauer, M. 2018. Understanding Student Procrastination via Mixture Models. *International Educational Data Mining Society* .

Perrin, C. J.; Miller, N.; Haberlin, A. T.; Ivy, J. W.; Meindl, J. N.; and Neef, N. A. 2011. Measuring and Reducing Colledge Students' Procrastination. *Journal of applied behavior analysis* 44(3): 463–474.

Shang, J.; and Sun, M. 2018. Local low-rank Hawkes processes for modeling temporal user–item interactions. *Knowledge and Information Systems* 1–24.

Shang, J.; and Sun, M. 2019. Geometric Hawkes Processes with Graph Convolutional Recurrent Neural Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 4878–4885.

Steel, P. 2007. The nature of procrastination: A meta-analytic and theoretical review of quintessential self-regulatory failure. *Psychological bulletin* 133(1): 65.

Uzir, N. A.; Gašević, D.; Jovanović, J.; Matcha, W.; Lim, L.-A.; and Fudge, A. 2020. Analytics of time management and learning strategies for effective online learning in blended environments. In *Proceedings of the Tenth International Conference on Learning Analytics & Knowledge*, 392–401.

Vassøy, B.; Ruocco, M.; de Souza da Silva, E.; and Aune, E. 2019. Time is of the essence: a joint hierarchical rnn and point process model for time and item predictions. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 591–599.

Vitiello, M.; Walk, S.; Helic, D.; Chang, V.; and Guetl, C. 2018. User Behavioral Patterns and Early Dropouts Detection: Improved Users Profiling through Analysis of Successive Offering of MOOC. *J. UCS* 24(8): 1131–1150.

Xiao, S.; Yan, J.; Yang, X.; Zha, H.; and Chu, S. M. 2017. Modeling the intensity function of point process via recurrent neural networks. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Yao, M.; Sahebi, S.; and Feyzi-Behnagh, R. 2020. Analyzing Student Procrastination in MOOCs: A Multivariate Hawkes Approach. In *The 13th International Conference on Educational Data Mining*.

Zhou, K.; Zha, H.; and Song, L. 2013. Learning social infectivity in sparse low-rank networks using multi-dimensional hawkes processes. In *Artificial Intelligence and Statistics*, 641–649.