Ripple: Concept-Based Interpretation for Raw Time Series Models in Education

Mohammad Asadi[1], Vinitra Swamy[1], Jibril Frej[1], Julien Vignoud[1], Mirko Marras[2], Tanja Käser[1]

[1] EPFL
[2] University of Cagliari
mohammad.asadi@epfl.ch, vinitra.swamy@epfl.ch, jibril.frej@epfl.ch,
julien.vignoud@epfl.ch, mirko.marras@acm.org, tanja.kaser@epfl.ch

Abstract

Time series is the most prevalent form of input data for educational prediction tasks. The vast majority of research using time series data focuses on hand-crafted features, designed by experts for predictive performance and interpretability. However, extracting these features is labor-intensive for humans and computers. In this paper, we propose an approach that utilizes irregular multivariate time series modeling with graph neural networks to achieve comparable or better accuracy with raw time series clickstreams in comparison to hand-crafted features. Furthermore, we extend concept activation vectors for interpretability in raw time series models. We analyze these advances in the education domain, addressing the task of early student performance prediction for downstream targeted interventions and instructional support. Our experimental analysis on 23 MOOCs with millions of combined interactions over six behavioral dimensions show that models designed with our approach can (i) beat state-of-the-art educational time series baselines with no feature extraction and (ii) provide interpretable insights for personalized interventions. Source code: https://github.com/epfl-ml4ed/ripple/.

Introduction

Over the last three years, there has been a 10-fold increase in digital learners on massive open online courses (MOOCs), contributing to a popular and data-rich setting in education (Impey and Formanek 2021; Shah 2021). In enabling a completely online learning experience, MOOCs suffer from high dropout and low success rates (Aldowah et al. 2020). Thus, an important task to counter these phenomena is providing personalized guidance at scale (Perez-Sanagustin et al. 2021). This task requires (i) predicting student performance early enough to intervene and adjust learning pathways and (ii) interpreting which behavior contributes to failing and passing trajectories for each student.

There exists a large body of approaches on student success prediction in MOOCs, e.g., random forests (Marras, Vignoud, and Kaser 2021; Sweeney et al. 2016), logistic regression (Whitehill et al. 2017), or neural networks (Wang et al. 2017; Mubarak, Cao, and Ahmed 2021). Most of these methods operate post-hoc, i.e., in the context of the entire time series. Only few works have focused on predicting success early on during the course. For example, Mbouzao, Desmarais, and Shrier (2020) predicted students' pass-fail grades based on video interactions, while Mao (2019) used temporal patterns to intervene early in programming tasks. Most of the work has employed hand-crafted expert-designed features, ranging from engagement-based features, such as course attendance rates (He et al. 2018) or the number of online sessions (Chen and Cui 2020; Lemay and Doleck 2020), to features capturing fine-grained video behavior (Akpinar, Ramdas, and Acar 2020; Mubarak, Cao, and Ahmed 2021) and measuring students' learning regularity (Boroujeni et al. 2016). Recently, Marras, Vignoud, and Kaser (2021) performed a meta-analysis on early success prediction features, showing that their predictive power does not often generalize across courses and raising questions about which features should be selected based on the course characteristics. Designing and extracting features for educational time series hence becomes expensive in terms of human and computational resources. Minimal literature has addressed raw time series in education. Prenkaj et al. (2021) used auto-encoders for risk prediction in MOOCs, but did not provide comparisons to hand-crafted baselines.

Using raw time series in combination with neural networks has also led to black-box models. In response to this issue, there has been a strong increase in research on neural network explainability, with methods such as LIME (Ribeiro, Singh, and Guestrin 2016), SHAP (Lundberg and Lee 2017), and counterfactual explanations (Dhurandhar et al. 2018). Only few works have however focused on explainability in the domain of education. Prior research has used LIME to provide local explanations for performance prediction models (Hasib et al. 2022; Vultureanu-Albişi and Bădică 2021) or to build a basis for students dashboards (Scheers and De Laet 2021). Baranyi, Nagy, and Molontay (2020) applied SHAP to interpret student dropout prediction models. However, a major shortcoming of those methods is that they do not seem to agree about what features are important in MOOCs (Swamy et al. 2022). Furthermore, interpretations are limited only to the engineered features originally during model training. On raw time series predictions, the minimal existing literature has shown attention heatmaps for temporal insights, not higher level, human-friendly actionable features for educator interventions (Ismail et al. 2020).

In this paper, we propose `Ripple` (Raindrop InterPretability PipeLine for Education), a novel methodology for providing interpretable early student success prediction using raw time series data. In contrast to prior work, our pipeline does not require any feature engineering, while still providing accurate predictions as well as human-friendly explanations. Our pipeline is based on the combination of a graph-based neural network approach (Zhang et al. 2021) for classifying raw time series of student interactions and the adaptation of concept activation vectors (TCAV) (Kim et al. 2018) for interpreting the neural network's internal state. Specifically, we use six well-defined dimensions of self-regulated learning in online courses from recent literature (Mejia-Domenzain et al. 2022) to provide interpretability in the global and local context. To the best of our knowledge, TCAV has never been applied on time series. We evaluate our pipeline on a large educational data set including 23 MOOCs with over $100,000$ students and millions of interactions, addressing the following research questions:

1. Can we use raw time series as input and achieve comparable performance to hand-crafted features?
2. Can we obtain interpretability on raw multivariate time series through learner-centric concept activation vectors?

Our results show that graph neural networks allow us to achieve comparable or better performance with raw time series models to hand-crafted features in 18 out of 23 courses and beat other state-of-the-art time series baselines on 21 out of 23 courses. Moreover, we showcase our interpretable pipeline on a selected digital signal processing course.

Methodology

This paper targets a classification task that utilizes raw multivariate time series to predict student pass-fail labels *early* in a course. Our goal is to achieve at least comparable performance using raw time series data in comparison to hand-crafted features (e.g., Marras, Vignoud, and Kaser (2021)), without compromising on interpretability. We first formalize the posed problem and then describe our methodology.

Problem Formalization

Given a course c part of the offering $\mathbb{C}$, we denote as $\mathbb{S}^c$ the set of students enrolled in c. Since each course can be run multiple times, we define a course set $\tilde{\mathbb{C}} = \{c_1, \ldots, c_{M^{\tilde{\mathbb{C}}}}\} \subset \mathbb{C}$ as the set of all iterations of the same course over the years, with $M^{\tilde{\mathbb{C}}}$ being the total number of iterations for the course set $\tilde{\mathbb{C}}$. Each course c includes a set of N_c learning objects denoted as $\mathbb{O}^c$. Students interact with learning objects in $\mathbb{O}^c$. The interactions of a student $s \in \mathbb{S}^c$ are modeled as a time series $\mathbb{I}_s^c = \{i_1, i_2, \ldots\}$. Each interaction is represented with a tuple composed of a timestamp t, an action a, a learning object $o \in \mathbb{O}^c$, and optional metadata m, i.e., $i = (t, a, o, m)$. We denote as $y_s^c \in \{0, 1\}$ the pass-fail label for student s in course c. Training a classification model $\mathcal{M}_\theta : \mathbb{I} \to \{0, 1\}$ is an optimization problem aimed to minimize the expectation on the following objective function:

$$\tilde{\mathcal{M}_\theta} = \underset{\mathcal{M}_\theta}{\operatorname{argmin}} \underset{s \in \mathbb{S}^c}{\mathbb{E}} |\mathcal{M}_\theta(\mathbb{I}_s^c) - y_s^c| \tag{1}$$

To preserve transparency, we assume that the prediction $\tilde{y}_s^c = \tilde{\mathcal{M}_\theta}(\mathbb{I}_s^c)$ for student s in course c can be interpreted based on a set $\mathbb{P}$ of human-understandable educational concepts. Each concept $p \in \mathbb{P}$ is associated to a relative concept importance score $d^{c,p,y}$ ranging in $[0, 1]$. A value close to 0 (1) means that the concept p has a low (high) importance for the y-class model predictions based on the interactions $\mathbb{I}^c$. An example concept is student's *regularity* in the course.

Following this formalization, we devised our deep learning approach consisting of the three main stages illustrated in Fig. 1: (i) data collection and preprocessing ($\mathbb{I}_s^c$), (ii) raw time series classification ($\tilde{\mathcal{M}_\theta}$), and (iii) concept-based interpretation ($\mathbb{P}$). We discuss each stage in more detail.

Raw Time Series Collection and Preprocessing

Collection. We collected clickstream data involving interactions $\mathbb{I}$ for students $\mathbb{S}^c$ from MOOCs $c \in \mathbb{C}$, modelled as an irregular multivariate time series. We refer to our time series as irregular due to the non-uniform time interval the data was generated (e.g., a student did not interact for over a week). Multivariate refers to the learning objects involved in the actions $a \in \mathbb{A}$, used to model the time series $\mathbb{I}$.

To model each interaction $i = (t, a, o, m)$, we considered learning objects $\mathbb{O}^c$ of type video and problems, with the video actions $\mathbb{A}_v \subset \mathbb{A}$ = {Download, Error, Load, Pause, Play, Seek, SpeedChange, Stalled} and the problem actions $\mathbb{A}_p \subset \mathbb{A}$ = {IsAssignment, IsQuiz}. An ID was assigned to each video and problem. For each problem, the number of times it was attempted by the student (Problem SubmissionNum) was also tracked. Each timestamp $t \in \mathbb{N}$ and action $a \in \mathbb{A}$, alongside the metadata m of Video ID, Problem ID, and Problem SubmissionNum, were treated as separate variables in our time series. For brevity, we will refer to our irregular multivariate time series as simply raw time series.

Early-Dropout Filtering. A common archetype of MOOC student is a learner who watches only a few videos or makes only a few initial interactions ($\mathbb{I}_s^c$). Motivations for this behavior include misaligned expectations of course material, unexpected life circumstances, or intellectual curiosity for a small subset of videos (Onah, Sinclair, and Boyatt 2014; Goopio and Cheung 2021). These students can be easily predicted with fail labels ($y_s^c = 1$) by considering their (lack of) initial graded assignments using a simple logistic regression model. It does not make sense to pass this student subset to complex neural networks when they could be so concisely and accurately identified as failing without further analysis. To let our raw time series model focus on hard-to-identify students, we removed students which can be predicted as failing with 99% accuracy using two weeks of assignment data, via the same model proposed in (Swamy, Marras, and Käser 2022; Swamy et al. 2022)[1]. In the rest of this paper, we will refer to $\mathbb{S}^c$ as the set of students after this filtering.

Early Prediction Level Definition. To enable our model to support instructors during the course (Borrella, Caballero-Caballero, and Ponce-Cueto 2021; Xing and Du 2019; Whitehill et al. 2015), we considered an *early* prediction setting. Under an early prediction level $e \in [0, 100]$ rep-

[1]These models were optimized through a grid-search to determine the optimal accuracy threshold and number of weeks.

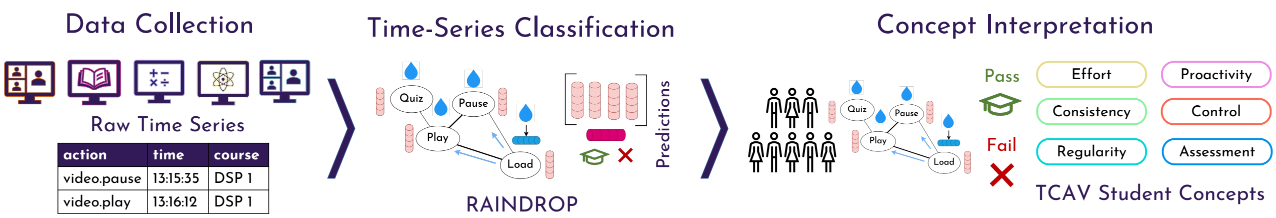


Figure 1: Our `Ripple` time series interpretability approach from logs collection to concept vector analysis.

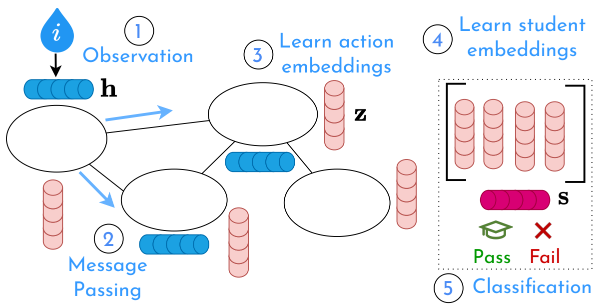


Figure 2: `Raindrop` time series classification approach for educational data: (1) observe an action and obtain interaction embeddings **h**; (2) use message passing to compute interaction embeddings for the unobserved actions; (3) learn action embeddings **z** from the interaction embeddings; (4) learn student embeddings **s** from the action embeddings; (5) course failure classification from student embeddings.

resenting the percentage of the course duration at which the prediction is delivered, we considered interaction data only up to that point in time. For instance, if $e = 60\%$ and the course lasts 10 weeks, we would consider only interactions happening in the first six course weeks. We denote the interactions of s in c up to the early prediction level e as $\mathbb{I}_s^{c,e}$.

Raw Time Series Classification

Motivation. The irregularity and multivariate nature in our time series is generally hard to analyze using classical machine learning temporal models that assume fully (or regularly) observed fixed-size inputs (Ismail Fawaz et al. 2019). To counter these issues, a recent time series representation model (`Raindrop`) assumes that actions are dependent and leverages their hidden structure by using a directed weighted dependency graph (Zhang et al. 2021). When an action $a \in \mathbb{A}$ is observed within an interaction $i \in \mathbb{I}$, this model updates a's internal representation and uses the dependency graph to update those of the actions related to a. The intuition is that an action observed at timestamp t can imply how unobserved actions would behave; updating these unobserved actions can improve the time series modeling.

Task Definition. Given an irregularly sampled multivariate time series $\mathbb{I}^c$, where each sample $\mathbb{I}_s^c$ has multiple but not always observed actions and each action has a different number of observations, our model $\tilde{\mathcal{M}}_\theta$ first learns a function $\mathcal{F} : \mathbb{I}_s^c \rightarrow \mathbb{R}^*$ that maps $\mathbb{I}_s^c$ to a fixed-length representation $\mathbf{s}_s^c$ (student embedding) suitable for classification. Using learned $\mathbf{s}_s^c$, this model then predicts the label $\tilde{y}_s^c$. The learned representation captures temporal patterns of irregular observations and considers dependencies between actions.

Model Learning. To learn the representation $\mathbf{s}_s^c$ for student s in a course c, we implemented the `Raindrop` architecture described in Zhang et al. (2021), generating student-level embeddings using a hierarchical architecture composed of three levels aimed to model interactions, actions, and students (see Fig. 2). First, we built a dependency graph $\mathcal{G}_s$ for every student s, where nodes represent actions and directed edges indicate the relation (with a weight ranging in $[0, 1]$) between two actions. The edge weights were initialized to 1 and optimized student-wise and time-wise via message passing, starting from the node associated to the observed action.

When an interaction $i = (t, a, o, m)$ was fed into the model for student s at time t, the model first embedded the interaction for the observed action (i.e., the action whose value was recorded) in an interaction embedding **h** using a non-linear transformation of the input. In order to update the interaction embeddings for unobserved actions at timestamp t, a graph neural network was used on top of the dependency graph $\mathcal{G}_s$. Once the interactions embeddings were generated from all timestamps, temporal self attention was used to aggregate all interactions embeddings associated to a given action into a single fixed-size representation **z**. The student embedding $\mathbf{s}_s^c$ was obtained by concatenating all action embeddings **z**. The final classifier is a fully-connected network that received $\mathbf{s}_s^c$ and output the pass-fail label $\tilde{y}_s^c$.

Concept-Based Model Interpretation

Motivation. Recent deep learning models, like `Raindrop`, trade transparency for accuracy. However, social, ethical and legislative requirements prominently call for model transparency, especially in education (Webb et al. 2021; Conati, Porayska-Pomsta, and Mavrikis 2018). Identifying the possible reasons behind a predicted failure in addition to predicting it accurately is crucial for designing effective interventions. A popular approach to interpretability is the use of post-hoc explainability methods, which return importance scores in terms of the input features the model originally considered. However, these methods appear ineffective on models receiving raw time series. Because of this difficulty, there is a need to shift towards learner-centric concept explanations. With this in mind, we adopt Kim et al. (2018)'s human-friendly quantitative testing based on concept activation vectors (TCAV), which gives an interpretation of a neural network's internal state in terms of human-interpretable

Dimensions	Measures	Patterns
Effort	Total time online Total video clicks	Higher intensity Lower intensity
Consistency	Mean session duration Relative time online Relative video clicks	Uniform First half Second half
Regularity	Periodicity of week day Periodicity of week hour Periodicity of day hour	Higher peaks Lower peaks
Proactivity	Content anticipation Delay in lecture view	Anticipated Delayed
Control	Fract. time spent (video) Pause action frequency Average change rate	Higher intensity Lower intensity
Assessment	Competency strength Student shape	Higher intensity Lower intensity

Table 1: Learning dimensions from Mejia-Domenzain et al. (2022) used as concepts for interpretability in our study.

concepts not explicitly considered as an input feature by the model. Used primarily for image and occasionally text data, this technique has never been used on (educational) time series input, to the best of our knowledge. The strengths of TCAV lay in its flexibility to analyze whichever concepts an educator finds pertinent for their course setting.

Concept Design and Extraction. To extract concepts for our educational scenario, we used the six learning dimensions (see Table 1) proposed by Mejia-Domenzain et al. (2022) due to the similarity of the underlying course data, the ease in interpreting these profiles, the clear identification of actionable insights based on patterns in these dimensions, the underpinning educational theory validated in them, and their relationship with academic performance. Concerning *effort*, *control*, and *assessment*, Mejia-Domenzain et al. (2022) found differences among profiles in terms of intensity (higher / lower). The *consistency* dimension was found to capture differences in the relative intensity over the course, with the majority of students having small peaks (uniform) and only a few students working more in the first/last course weeks (first / last half). Regarding *regularity*, some students were found to regularly work on specific weekdays (higher peaks), while others did not have a clear pattern (lower peaks). For *proactivity*, most students were found to interact with the content in advance (anticipated), whereas a minority often interacted with it after the deadline (delayed).

We derived our set of concepts $\mathbb{P}$ from Mejia-Domenzain et al. (2022)'s findings, using the above patterns emerged per dimension. For each dimension, we identified two or three student patterns (e.g., the subset of students showing the highest *effort* and the subset of students showing the lowest *effort*) and devised a greedy optimization protocol to select approximately 100 students that most fit the considered pattern. We achieve this by extracting the $t = 5\%$ of top students showing the considered student pattern for each *corresponding measure* in that dimension (see Table 1) and computing the intersection of these measure subsets. We then incrementally increased the threshold t until each combined pattern subset had at least 100 students[2].

Concept Importance Computation. For a given dimension, the two identified student subsets were given as an input to TCAV, which relied on them to (i) identify a hyperplane that best differentiates between the model activations produced by the subset and the activations in any model layer, and (ii) specify a CAV, i.e., the direction orthogonal to this hyperplane. Using the CAV directional derivative, we identified the importance score of each concept for the predictions our model returned. Formally, let y and $\mathbb{S}_y$ represent the pass-fail label and the set of students with that label respectively, and let $\mathcal{D}^{p,l}$ be the directional CAV derivative function for concept $p \in \mathbb{P}$ at the model layer $l \in \mathbb{L}$. The TCAV importance score for p is the fraction of y-class students whose activation vector was on average positively impacted by p:

$$\mathrm{d}^{c,p,y} = \frac{1}{|\mathbb{L}|} \sum_{l \,\in\, \mathbb{L}} \frac{\left|\left\{s \in \mathbb{S}_y : \mathcal{D}^{p,l}(s) > 0\right\}\right|}{|\mathbb{S}_y|}$$

TCAV importance scores range between $[0, 1]$. Higher values indicate that concept p has a high importance for the prediction of class y. The sensitivity of concepts to predictions can be specified for a population of students (global interpretation) or for individual students (local interpretation).

Experimental Evaluation

We examined whether models using raw time series as input can achieve comparable performance to models receiving hand-crafted features (RQ1) and whether we can obtain interpretable concept activation vectors to gain insights into model predictions (RQ2). In the following, we describe the dataset, optimization protocol, and the experiments in detail.

Dataset. Our dataset consisted of 23 MOOCs and $134{,}699$ students. Its entries are fully anonymized and correspond to courses offered by an European university worldwide between 2013 and 2015. Facets of this dataset have been used in educational machine learning work, such as Swamy, Marras, and Käser (2022); Swamy et al. (2022); Mejia-Domenzain et al. (2022); Li et al. (2015); Boroujeni and Dillenbourg (2018). The dataset records include fine-grained video and quiz interactions for each student, e.g., pressing pause on a video or submitting a quiz. After early-dropout filtering, our data set included $73{,}042$ students in total. The 23 courses were selected from a set of larger MOOC courses for diversity in topic, duration, level, language, and student population, which allowed us to provide a realistic estimation of model performance. The course size ranges from 452 to $11{,}151$ students. Table 2 lists detailed course information.

Optimization Protocol. To answer our research questions, we compared the performance of our model to the optimal bidirectional LSTM (BiLSTM) architecture using hand-crafted features, both described in Swamy, Marras, and

[2]We have experimentally validated that TCAV is not consistent or robust with less than 100 examples.

Course Title	Identifier	Field[1]	It.	No. Stud.[2]	Level	Lang.	No. Weeks	Passing Rate[3]	No. Quiz.[4]
CPP Programming	*CPP*	CS	2	1,517	Prop.	En/Fr	8/10	(38, 63)	12
Digital Signal Processing	*DSP*	CS	5	15,394	MSc	English	10	(17, 24)	38
Functional Programming	*ProgFun*	CS	2	18,702	BSc	French	7	(52, 82)	3
Analyse Numérique	*AnNum*	Math	3	1,468	BSc	French	9	(9, 75)	36
Éléments de Géomatique	*Geomatique*	Math	1	452	BSc	French	11	45	27
Household Water Treatment	*HWTS*	NS	2	2,423	BSc	French	5	(46, 49)	10
Microcontrôleurs	*Micro*	Eng	4	7,503	BSc	French	10	(8, 49)	18
Launching New Ventures	*Venture*	Bus	1	3,208	BSc	English	7	3	13
Villes Africaines	*VA*	SS	3	10,094	BSc/Prop.	En/Fr	12	(8, 11)	18

[1]**Field.** *Bus*: Business; *CS*: Computer Science; *Eng*: Engineering; *Math*: Mathematics; *NS*: Natural Science; *SS*: Social Science.
[2]**No. Students** is calculated after filtering out the early-dropout students, as detailed in the *Time Series Preprocessing* section.
[3]**Passing Rate** is the (min, max) of passing rate percentage over iterations. [4]**No. Quizzes** is the average number of quizzes.

Table 2: Detailed information on the MOOCs highlighted in our experiments.

Käser (2022). To provide another point of comparison, we also implemented the Set Functions for Time Series (SeFT) (Horn et al. 2020) and Transformers (Vaswani et al. 2017) baselines analyzed by Zhang et al. (2021) as other state-of-the-art models in the medical domain. We trained each model on the 23 course iterations listed in Table 2 under two early prediction levels ($e \in \{40\%, 60\%\}$). Following prior work (Swamy, Marras, and Käser 2022), our choice of these two levels is motivated by the fact that *HWTS*, the shortest course, has only 5 weeks. We used a $80 : 10 : 10$ train-test-validation split, making sure to assign each student's time series uniquely in either train, test, or validation. We monitored balanced accuracy (BAC) due to the high class imbalance[3]. For each model, the hyper-parameters were tuned via a grid search (please refer to our source code).

RQ1: Raw Time Series Classification

In a first analysis, we compared `Raindrop`'s performance to (i) state-of-the art models (Transformers, SeFT) using raw time series and (ii) a BiLSTM using 42 features engineered for educational data. Table 3 lists the BAC for all the models and course types for both the 40% and 60% early prediction levels. For each course type, the BAC is averaged over the number of courses, weighted by the number of students. In our preliminary experiments, we found that LSTMs and autoencoders could not converge on raw time series input, always creating models with 50% balanced accuracy.

For the 40% early prediction level, `Raindrop` achieves a comparable or better BAC than the state-of-the-art model with engineered features for 7 out of 9 course types. Moreover, `Raindrop` is overall the best model for six course types. At the level of a single course, `Raindrop` performs equally or better than Transformers for 21 out of 23 courses and than SeFT for 21 out of 23 courses. Comparing to engineered features, `Raindrop` using raw time series exhibits a higher or comparable BAC for 18 out of 23 courses (higher BAC: 12 courses, comparable BAC: 6 courses). Note that comparable is defined as a less than 5% decrease in BAC. Given the effort required for engineering the features, we deem a small decrease in predictive performance acceptable.

We observe similar results for the 60% early prediction level. Again, using raw time series with `Raindrop` leads to a BAC comparable (or higher) than using engineered features for 7 out of 9 course types. Furthermore, `Raindrop` is the best model for seven course types. At the level of single courses, `Raindrop` has equal or better performance than Transformers and SeFT for 21 and 20 courses, respectively. When considering both 40% and 60% early prediction, using `Raindrop` leads to accuracy levels higher than the BiLSTM using engineered features on 6 out of the 9 course types. HWTS and Micro are the only two course types where BiLSTM with hand-crafted features outperforms models that use the raw time series.

To summarize, `Raindrop` *models with raw time series show comparable and oftentimes better performance than hand-crafted feature models across* 23 *course iterations.*

RQ2: Interpretability using TCAV

In a second analysis, we investigated the use of concept activation vectors (TCAV) to create learner-centric interpretations of raw time series models. We use the six dimensions of learning highlighted in Table 1 as concepts across our `Raindrop` model to interpret which aspects of the time series the model found important in determining student performance labels. In the following, we will examine these results on the DSP course at the 40% predictive level to provide a comparative analysis to the interpretability study in Swamy et al. (2022). This `Ripple` analysis can easily be extended to other courses or time series settings.

Figure 3 showcases TCAV concept sensitivity scores on DSP, across the model prediction classes of pass and fail. The significance of a specific pattern (i.e., *uniform consistency*) can be analyzed in comparison to a random concept, defined by randomly choosing a subset of 100 students without replacement 100 times. TCAV scores are computed relative to other concepts in the same plot, so a low random concept score indicates that the other concepts are particularly important. We note that *consistency* in the second half of the course for DSP is an important indicator of student

[3]We found metrics other than BAC (e.g., F1, AUC, precision, and recall) to show a biased perspective of model performance.

	Early 40%							Early 60%						
	Raindrop	**SeFT**		**TF**		**BiLSTM**		**Raindrop**	**SeFT**		**TF**		**BiLSTM**	
	BAC	BAC	R	BAC	R	BAC	R	BAC	BAC	R	BAC	R	BAC	R
CPP*	**0.57**	0.46	2/2	0.54	2/2	0.56	2/2	**0.55**	0.53	1/2	0.52	2/2	**0.55**	2/2
DSP*	**0.81**	0.72	5/5	0.59	5/5	0.80	4/5	**0.91**	0.82	5/5	0.62	5/5	**0.91**	4/5
ProgFun*	**0.76**	0.63	2/2	0.53	2/2	0.63	2/2	**0.75**	0.69	2/2	0.56	2/2	0.67	2/2
AnNum	**0.66**	0.51	3/3	0.51	3/3	0.62	3/3	0.55	0.57	3/3	0.51	3/3	**0.69**	1/3
Geomatique*	0.50	0.45	1/1	**0.56**	0/1	0.47	1/1	**0.77**	0.55	1/1	0.45	1/1	0.76	1/1
HWTS	0.61	0.55	2/2	0.55	1/2	**0.71**	1/2	0.62	0.62	1/2	0.56	2/2	**0.73**	0/2
Micro	0.74	0.70	2/4	0.58	4/4	**0.81**	1/4	**0.78**	0.76	2/4	0.63	2/4	**0.78**	2/4
Ventures*	**0.77**	0.64	1/1	0.64	1/1	0.50	1/1	**0.88**	0.73	1/1	0.56	1/1	0.60	1/1
VA*	**0.88**	0.75	3/3	0.63	3/3	0.80	3/3	**0.90**	0.72	3/3	0.68	3/3	0.83	3/3

The best model for each course type and early prediction level is marked in **bold**. Course types where `Raindrop` had comparable or better performance to BiLSTM on both early prediction levels are marked in (*).

Table 3: Performance comparison between three raw time series models (Raindrop, SeFT, Transformers) and a hand-crafted feature-based model (BiLSTM). Balanced Accuracy (BAC) is averaged over iterations of the same course and weighted by the number of students. R indicates the proportion of iterations of a course where `Raindrop` outperforms the baseline.

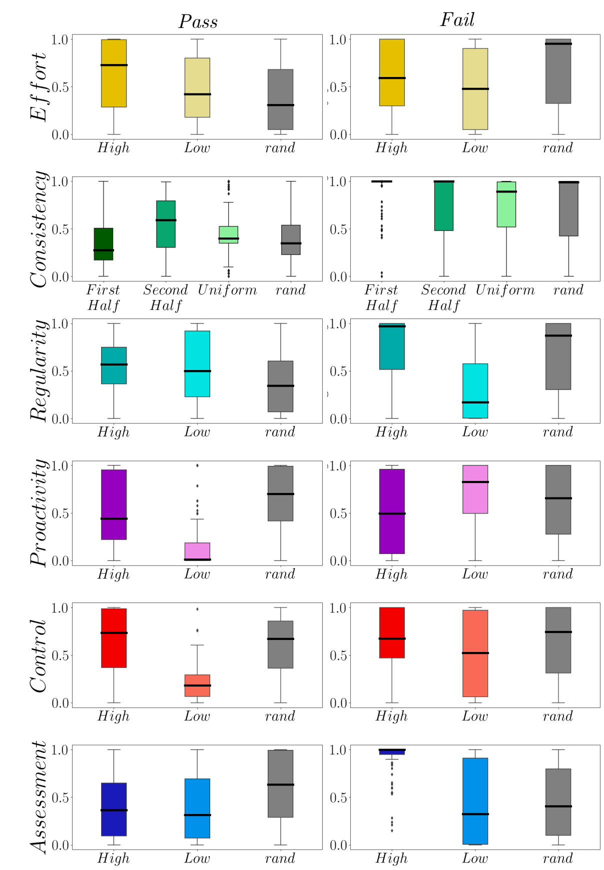


Figure 3: TCAV plots for early 40% prediction on DSP to determine the importance of a concept to the model's predictions. For example, the last row shows a high TCAV score for high *assessment* and a low TCAV score for random concepts for predicting student failure. This hints that the model is sensitive to *assessment* for predicting failure.

success, more than consistent behavior in the first half or uniform consistency through the course. Interestingly, the model is sensitive to high *assessment* scores when predicting student failure. Note that the score represents a sensitivity of the model only, with no indication on its direction. We hypothesize that performing well in assessments is actually a good indication for **not** failing, which is confirmed by the distribution of scores for *assessment* measures when the model predicts failure. We further observe that the model is sensitive to high *effort* when predicting passing.

While this analysis provides a global perspective, it is also important to identify local, actionable insights based on early predictions. Our `Ripple` TCAV formulation enables the pipeline to provide interpretations for individual students. To observe local explanations, we examined the TCAV scores across a few interesting dimensions[4] for a high performer (student A) and a low performer (student B). A student's performance was measured based on the average of all of the six dimensions from Table 1, i.e. a highly performant student is among the top $t\%$ of all the dimensions combined and vice versa. The motivation for choosing these two case studies is two-fold: (i), we aimed to showcase a real-world use case for an educator to make individual student interventions, and (ii), we wanted to validate our interpretability methods on students who should have very different scores on TCAV dimensions because of their different levels of engagement. We computed TCAV plots across all behavioral dimensions for DSP on these two students and highlight three dimensions with interesting results in Figure 4. We saw that *consistency* is indicative of performance for the high performer in the second half of the course and for the low performer in the first half of the course. We saw that low *regularity* is a trait of the high performer while neither high nor low *regularity* is important for student B (the random concept has a high TCAV score). Lastly, while *assessment* is not important to student A, low *assessment* score indicates the failing prediction of student B.

We are also interested in using TCAV to build intuition about how and why the model makes mistakes. Deep learn-

[4]Extended `Ripple` results can be found in our repository.

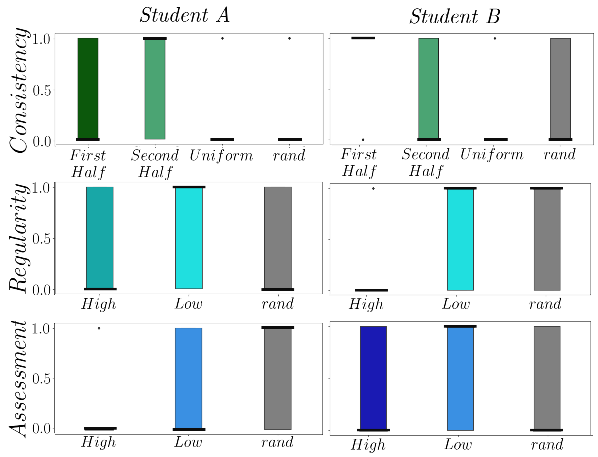
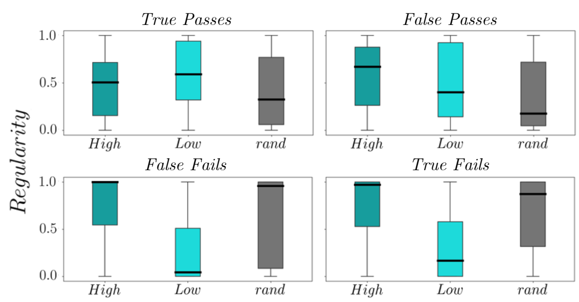
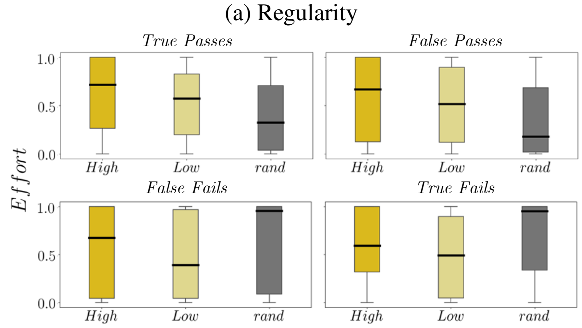


Figure 4: TCAV score plots for two students with differing behavioral characteristics in DSP. Student A is a high performer and student B is a low performer.



(a) Regularity



(b) Effort

Figure 5: Confusion matrix of TCAV plots across two dimensions for early 40% prediction on DSP. *True* or *False* designate the model's correctness in prediction; *True Passes* indicates the model predicted pass for students who did pass.

ing models have not seen the same uptick in adoption as traditional machine learning methods due to their lack of interpretability, which demonstrates practitioners' mistrust in model prediction. We aim to use TCAV to help educators trust time series prediction models by providing an avenue to understand model strengths and weaknesses.

In Figure 5, we examined *regularity* and *effort* for DSP. Specifically, we investigated the four cases present in a traditional confusion matrix analysis, observing when the ground truth agrees with the model's predictions (true positives and true negatives) and when they disagree (false positives, false negatives). For *regularity* (Figure 5a), we observed an inverse relationship between true positives and false positives. When the model made a correct prediction for passing students, it was more sensitive to low *regularity* and when the model predicted a false positive, it was more sensitive to high *regularity*. We can hypothesize that when `Ripple` incorrectly identified a student as passing, this student has high *regularity* scores and that tricked the model into getting it wrong. For false and true negatives, we see high random concept values for *regularity*. We can infer that high and low *regularity* are not important concepts for predicting failure. Similarly, in Figure 5b for the *effort* dimension, we can infer that neither high nor low *effort* concepts contributed to predicting failure. However, for the passing case, we see a parallel relationship across true and false positives: the model likely always predicted that students with high *effort* pass the course (and sometimes this was incorrect). Going further to examine the distributions of the dimension values for each of these student subsets would enable educators to validate these hypotheses. Through confusion matrix plots, it is possible to examine different subsets of the student population to analyze model strengths and weaknesses in detail.

Overall, `Ripple` *enables globally interpretable feedback on the scale of thousands of students in a course and locally actionable feedback on the scale of a specific student without requiring a model built on hand-crafted features.*

Conclusion

In this paper, we introduced `Ripple`, a pipeline to make predictions from raw time series and interpret them with learner-centric concepts. We demonstrated that the performance of the underlying `Raindrop` models is comparable and often considerably better than hand-crafted feature models. Furthermore, we showed that it is feasible to define human-friendly concepts and make intuitive and actionable interpretations of model behavior. We also suggested the use of TCAV interpretability analysis to build trust in models.

The novelty of this work lies in combining the state-of-the-art AI advances in time series modeling and interpretability together and examining their implications for educational data. Educators can now get granular insights about their course (global scale) and individual students (local scale) that are based on concepts that they specify as important. The flexibility of interpretation that TCAV offers (user-specified concepts, granularity of insights, accuracy in directly using model activations) applies to any educational time series prediction setting and beyond, making it ideal for any scenario where there is direct impact on humans.

In future work, we plan to run experiments on a larger dataset, with a more international audience and more interaction modalities (i.e., flipped classrooms, simulation data). We also hope to extend `Ripple` using transfer learning across courses and to provide generalized concept vectors to be used for a multi-course model. This would allow to use raw time series input with deep learning models and maintain both accuracy and interpretability without extra effort. Finally, `Raindrop` can be further optimized for efficiency.

References

Akpinar, N.; Ramdas, A.; and Acar, U. 2020. Analyzing Student Strategies In Blended Courses Using Clickstream Data. In *Proceedings of the 13th International Conference on Educational Data Mining*, 6–17.

Aldowah, H.; Al-Samarraie, H.; Alzahrani, A. I.; and Alalwan, N. 2020. Factors affecting student dropout in MOOCs: a cause and effect decision-making model. *Journal of Computing in Higher Education*, 32(2): 429–454.

Baranyi, M.; Nagy, M.; and Molontay, R. 2020. Interpretable deep learning for university dropout prediction. In *Proceedings of the 21st Annual Conference on Information Technology Education*, 13–19.

Boroujeni, M. S.; and Dillenbourg, P. 2018. Discovery and temporal analysis of latent study patterns in MOOC interaction sequences. In *Proceedings of the 8th International Conference on Learning Analytics and Knowledge*, 206–215.

Boroujeni, M. S.; Sharma, K.; Kidziński, Ł.; Lucignano, L.; and Dillenbourg, P. 2016. How to quantify student's regularity? In *Proceedings of the European Conference on Technology Enhanced Learning*, 277–291. Springer.

Borrella, I.; Caballero-Caballero, S.; and Ponce-Cueto, E. 2021. Taking action to reduce dropout in MOOCs: Tested interventions. *Computers & Education*, 104412.

Chen, F.; and Cui, Y. 2020. Utilizing Student Time Series Behaviour in Learning Management Systems for Early Prediction of Course Performance. *Journal of Learning Analytics*, 7(2): 1–17.

Conati, C.; Porayska-Pomsta, K.; and Mavrikis, M. 2018. AI in Education needs interpretable machine learning: Lessons from Open Learner Modelling. *Proceedings of the ICML Workshop on Human Interpretability in Machine Learning*.

Dhurandhar, A.; Chen, P.-Y.; Luss, R.; Tu, C.-C.; Ting, P.; Shanmugam, K.; and Das, P. 2018. Explanations based on the missing: Towards contrastive explanations with pertinent negatives. *Advances in Neural Information Processing Systems*, 31.

Goopio, J.; and Cheung, C. 2021. The MOOC dropout phenomenon and retention strategies. *Journal of Teaching in Travel & Tourism*, 21(2): 177–197.

Hasib, K. M.; Rahman, F.; Hasnat, R.; and Alam, M. G. R. 2022. A Machine Learning and Explainable AIApproach for Predicting Secondary School Student Performance. In *Proceedings of the IEEE 12th Annual Computing and Communication Workshop and Conference*, 0399–0405.

He, H.; Zheng, Q.; Dong, B.; and Yu, H. 2018. Measuring Student's Utilization of Video Resources and Its Effect on Academic Performance. In *Proceedings of the 18th IEEE International Conference on Advanced Learning Technologies, ICALT 2018*, 196–198. IEEE Computer Society.

Horn, M.; Moor, M.; Bock, C.; Rieck, B.; and Borgwardt, K. 2020. Set Functions for Time Series. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119, 4353–4363. PMLR.

Impey, C.; and Formanek, M. 2021. MOOCS and 100 Days of COVID: Enrollment surges in massive open online astronomy classes during the coronavirus pandemic. *Social Sciences & Humanities Open*, 4(1): 100177.

Ismail, A. A.; Gunady, M.; Corrada Bravo, H.; and Feizi, S. 2020. Benchmarking Deep Learning Interpretability in Time Series Predictions. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 6441–6452. Curran Associates, Inc.

Ismail Fawaz, H.; Forestier, G.; Weber, J.; Idoumghar, L.; and Muller, P.-A. 2019. Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery*, 33(4): 917–963.

Kim, B.; Wattenberg, M.; Gilmer, J.; Cai, C.; Wexler, J.; Viegas, F.; et al. 2018. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *Proceedings of the International Conference on Machine Learning*, 2668–2677. PMLR.

Lemay, D. J.; and Doleck, T. 2020. Grade prediction of weekly assignments in MOOCS: mining video-viewing behavior. *Education and Information Technologies*, 25(2): 1333–1342.

Li, N.; Kidzinski, L.; Jermann, P.; and Dillenbourg, P. 2015. How do in-video interactions reflect perceived video difficulty? Technical report, PAU Education.

Lundberg, S. M.; and Lee, S.-I. 2017. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30.

Mao, Y. 2019. One minute is enough: Early prediction of student success and event-level difficulty during novice programming tasks. In *Proceedings of the 12th International Conference on Educational Data Mining*.

Marras, M.; Vignoud, J. T. T.; and Kaser, T. 2021. Can feature predictive power generalize? benchmarking early predictors of student success across flipped and online courses. In *Proceedings of the 14th International Conference on Educational Data Mining*, 150–160.

Mbouzao, B.; Desmarais, M. C.; and Shrier, I. 2020. Early Prediction of Success in MOOC from Video Interaction Features. In *Proceedings of the 21st International Conference on Artificial Intelligence in Education, AIED 2020*, volume 12164 of *Lecture Notes in Computer Science*, 191–196. Springer.

Mejia-Domenzain, P.; Marras, M.; Giang, C.; and Käser, T. 2022. Identifying and Comparing Multi-dimensional Student Profiles Across Flipped Classrooms. In *Proceedings of the International Conference on Artificial Intelligence in Education*, 90–102. Springer.

Mubarak, A. A.; Cao, H.; and Ahmed, S. A. M. 2021. Predictive learning analytics using deep learning model in MOOCs' courses videos. *Education and Information Technologies*, 26(1): 371–392.

Onah, D. F.; Sinclair, J.; and Boyatt, R. 2014. Dropout rates of massive open online courses: behavioural patterns. *Proceedings of the 6th International Conference on Education and New Learning Technologies*, 1: 5825–5834.

Perez-Sanagustin, M.; Pérez-Álvarez, R.; Maldonado-Mahauad, J.; Villalobos, E.; Hilliger, I.; Hernández, J.; Sapunar, D.; Moreno-Marcos, P. M.; Muñoz-Merino, P. J.; Delgado Kloos, C.; et al. 2021. Can Feedback based on Predictive Data Improve Learners' Passing Rates in MOOCs? A Preliminary Analysis. In *Proceedings of the Eighth ACM Conference on Learning @ Scale*, 339–342.

Prenkaj, B.; Distante, D.; Faralli, S.; and Velardi, P. 2021. Hidden space deep sequential risk prediction on student trajectories. *Future Generation Computer Systems*, 125: 532–543.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144.

Scheers, H.; and De Laet, T. 2021. Interactive and Explainable Advising Dashboard Opens the Black Box of Student Success Prediction. In *Proceedings of the European Conference on Technology Enhanced Learning*, 52–66. Springer.

Shah, D. 2021. By The Numbers: MOOCs in 2021. *Class Central*.

Swamy, V.; Marras, M.; and Käser, T. 2022. Meta Transfer Learning for Early Success Prediction in MOOCs. In *Proceedings of the Ninth ACM Conference on Learning@ Scale*, 121–132.

Swamy, V.; Radmehr, B.; Krco, N.; Marras, M.; and KÃ¤ser, T. 2022. Evaluating the Explainers: Black-Box Explainable Machine Learning for Student Success Prediction in MOOCs. In *Proceedings of the 15th International Conference on Educational Data Mining*, 98–109.

Sweeney, M.; Rangwala, H.; Lester, J.; and Johri, A. 2016. Next-term student performance prediction: A recommender systems approach. *arXiv preprint arXiv:1604.01840*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; and Polosukhin, I. 2017. Attention is All you Need. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Vultureanu-Albişi, A.; and Bădică, C. 2021. Improving Students' Performance by Interpretable Explanations using Ensemble Tree-Based Approaches. In *Proceedings of the IEEE 15th International Symposium on Applied Computational Intelligence and Informatics*, 215–220. IEEE.

Wang, L.; Sy, A.; Liu, L.; and Piech, C. 2017. Deep knowledge tracing on programming exercises. In *Proceedings of the 4th ACM Conference on Learning @ Scale*, 201–204.

Webb, M. E.; Fluck, A.; Magenheim, J.; Malyn-Smith, J.; Waters, J.; Deschênes, M.; and Zagami, J. 2021. Machine learning for human learners: opportunities, issues, tensions and threats. *Educational Technology Research and Development*, 69(4): 2109–2130.

Whitehill, J.; Mohan, K.; Seaton, D.; Rosen, Y.; and Tingley, D. 2017. MOOC dropout prediction: How to measure accuracy? In *Proceedings of the 4th ACM Conference on Learning @ Scale*, 161–164.

Whitehill, J.; Williams, J.; Lopez, G.; Coleman, C.; and Reich, J. 2015. Beyond prediction: First steps toward automatic intervention in MOOC student stopout. *Available at SSRN 2611750*.

Xing, W.; and Du, D. 2019. Dropout prediction in MOOCs: Using deep learning for personalized intervention. *Journal of Educational Computing Research*, 57(3): 547–570.

Zhang, X.; Zeman, M.; Tsiligkaridis, T.; and Zitnik, M. 2021. Graph-Guided Network for Irregularly Sampled Multivariate Time Series. In *Proceedings of the International Conference on Learning Representations*.