Behavior in Social Learning Networks: Early Detection for Online Short-Courses

Weiyu Chen*, Christopher G. Brinton*, Da Cao*, Mung Chiang†
*Advanced Research, Zoomi Inc. †Department of Electrical Engineering, Princeton University
*{weiyu.chen, christopher.brinton, da.cao}@zoomiinc.com †chiangm@princeton.edu

***Abstract*—We study learning outcome prediction for online courses. Whereas prior work has focused on semester-long courses with frequent student assessments, we focus on short-courses that have single outcomes assigned by instructors at the end. The lack of performance data makes the behavior of learners, captured as they interact with course content and with one another in Social Learning Networks (SLN), essential for prediction. Our method defines several (machine) learning features based on behaviors collected on the modes of (human) learning in a course, and uses them in appropriate classifiers. Through evaluation on data captured from three two-week courses hosted through our delivery platforms, we make three key observations: (i) behavioral data is predictive of learning outcomes in short-courses (our classifiers achieving AUCs ≥ 0.8 after the two weeks), (ii) it has an early detection capability (AUCs ≥ 0.7 with the first week of data), and (iii) the content features have an "earliest" detection capability (with higher AUC in the first few days), while the SLN features become the more predictive set over time, as the network matures. We also discuss how our method can generate behavioral analytics for instructors.**

I. Introduction

A multitude of online learning platforms have emerged over the past decade, offering services ranging from tutoring to professional development to higher education. For all its benefits, however, the quality of online learning has been criticized. In comparing it to traditional, face-to-face instruction, research has cited *e.g.,* lower engagement and knowledge transfer to learners, both in higher education [1] and corporate training [2]. These poorer outcomes have been attributed to factors such as the asynchronous nature of interaction online, which places limitations on social learning [3].

In free, open online courses, lower engagement and knowledge transfer may be acceptable, because learners have varying motivations for enrolling in the first place. Yet, in the case of corporate training, while well over $50 billion has been spent on training by US corporations each year since 2009, engagement, retention, and knowledge transfer to the workplace are not meeting the expectations of many employers [4].

A. Predictive Learning Analytics

Predictive Learning Analytics (PLA) is emerging as a research area with the promise of helping instructors improve course quality, particularly in online courses [3]. Prediction of *e.g.,* student drop-off rates [5], quiz scores [6], exam performance [7], and beneficial collaboration groups [8] each detect scenarios for which instructor intervention has a high chance of positively impacting the learning experience.

Most PLA methods have been developed for and evaluated on semester-long courses, *e.g.,* in higher education [7] and Massive Open Online Courses (MOOCs) [1]. These courses usually have frequent assessments to track student progress, which has been the most common form of data used in PLA models, through *e.g.,* matrix factorization to discover patterns across student scores [6]. But what about cases in which assessments are *not* used frequently, if at all? This is common in online corporate training and professional certification, with courses that may last only several days and have smaller enrollments [9]. Needed for these "short-courses" are PLA algorithms that rely on the forms of data that are available.

Today, online course delivery platforms can collect behavioral measurements about learners, which includes how they interact in Social Learning Networks (SLN) [3] and with the course content. The resulting content clickstream [1] and SLN [8] data present novel opportunities to design PLA methods that model learner attributes based on behavioral data. This paper presents and evaluates one such method for learning outcome prediction, using data captured from short-courses hosted with our course delivery Player, instructor Dashboard, and integrated discussion Forum.

B. Behavior-Based Outcome Prediction

In this work, we investigate the following research questions related to learning outcome prediction:

- *Can we use behavior alone to predict learning outcomes in short-courses?*
- *How early into short-courses can these predictions be made with reasonable quality?*
- *Is one type of learning behavior – with content or within social learning networks – more effective for prediction?*

Researchers have proposed algorithms for student performance prediction that augment assessment-based methods with behavior-based machine learning features [1], [6], [10]. Motivated by these schemes, in this work we consider the challenging case of short-courses without intermediary assessments, thereby necessitating fully behavior-based PLA.

Our methodology. Fig. 1 summarizes the main components of the methodology we develop in this paper. To make predictions during the nth day of a course's current offering, we use the behavioral data collected from the first n days of prior offerings of this course as input. Using our system architecture for data capture (summarized in Sec. II), one of the key challenges is to transform this raw data to effective feature

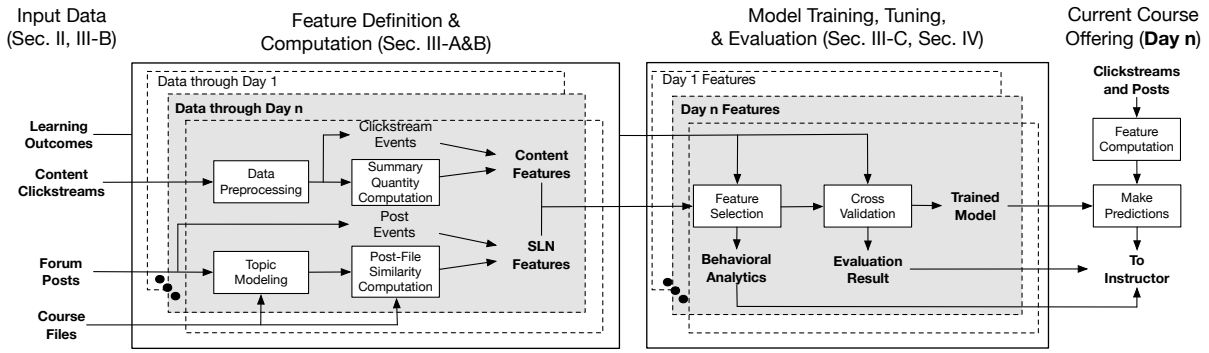


Fig. 1: Summary of the different components of the learning outcome prediction method we develop in this paper.

sets for modeling learning behavior, which we address in Sec. III-A. In particular, we define two types of features:
(i) Content features: These features summarize learner behavior while interacting with course content in the Player. They include a novel definition of "engagement" on learning modes.
(ii) SLN features: These features summarize learner discussions in the Forum. They include the similarity between a learner's posts and the different units of course content, determined through appropriate topic modeling.

Prior works applying content features to prediction [5], [6] have focused on clickstream data from a single learning mode, without an explicit engagement metric. On the other hand, works that have considered SLN features [10], [11] have neglected a topic similarity component. Our feature selection results (Sec. III-C) show that both of these components are correlated with outcomes in short-courses.

With the objective of predicting whether a learner will ultimately pass or fail, our method uses these feature sets as input to different classifiers in training and evaluation, described in Sec. IV. The choice of classifier, parameters, and coefficients is made through cross validation (Sec. IV-A). The evaluation result from this stage (Sec. IV-C), as well as behavioral analytics from the feature correlations (Sec. IV-D), can be shared with instructors to give them an indication of expected prediction quality and ways to assist learners. Finally, the real-time predictions and corresponding early detections are made by applying the trained model to the features computed on the data collected thus far in the current offering.
Evaluation and key results. To evaluate our outcome prediction method, we use datasets from three recent courses (described in Sec. III-B) we delivered for a professional training course provider in the US. Each course session lasts two weeks and has a single binary outcome (pass/fail) at the end that is determined by the instructor (see Table II). Through simulating the predictions for each course using our day-by-day modeling approach, we make three main observations:

- The highest performing algorithms reach $\geq 80\%$ AUC by the end of the courses, with $\leq 10\%$ Type II error.
- Using only the first week of data, the algorithms can still reach $\geq 70\%$ AUC, which underscores the early detection capability of behavioral data in short-courses.
- The content features exhibit an "earliest" detection capability in the first few days of a course, while the SLN features bring superior quality after that.

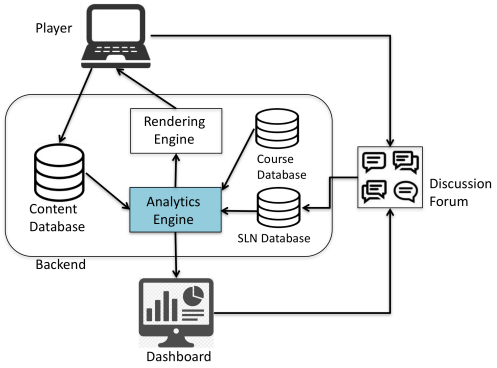


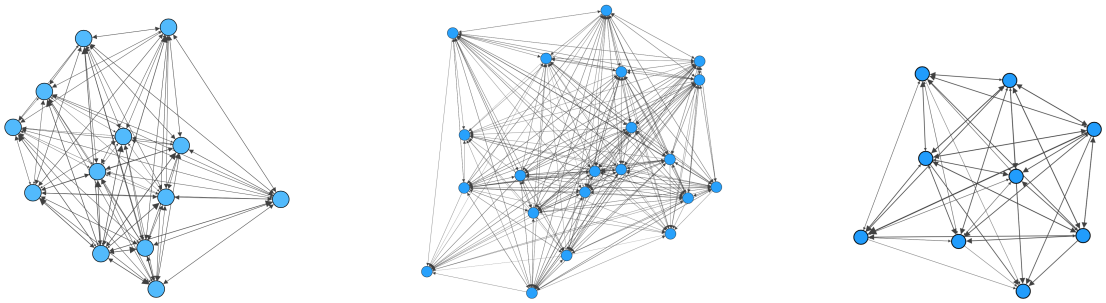
Fig. 2: Overview of our system architecture.

II. Summary of System Developed and Used

Our system has four main parts, shown in Fig. 2: the course delivery Player, the analytics Dashboard, the discussion Forum, and the Backend. We briefly describe the first three parts here; for more information and screenshots of the Player and Dashboard, see our technical report [12].

A. Player: Learner-facing

Learners obtain access to the Player through a web browser. The data measurements collected through the Player are used to compute the content-based learning features in Sec. III.
Course architecture. Each course is organized into a set of modules, each module consisting of one or more units. A unit is the most basic entity of a course, *i.e.,* the course is delivered as a sequence of units. Within each unit, a number of content learning modes may be available to the learner. These modes can include interactive slideshows (*e.g.,* Articulate Storyline), PDFs, text articles, and lecture videos, depending on what the instructor has offered for the learners. In the courses we consider in this paper, each unit is some combination of interactive slides, PDFs, and articles.
User functions and data capture. In interactive slideshows, a learner can perform the following actions: `play` (`Pl`), `pause` (`Pa`), and `skip` forward (`Sf`) or backwards (`Sb`) on the current slide, `replay` the current slide, and `advance` to the next slide. Within a PDF and an article, learner can `scroll` up (`Su`) or down (`Sd`) on the pages. Each time one of



(a) VE (b) ES (c) LS

Fig. 3: Graph of SLN on the discussion forums, for one session of each course analyzed in this paper (see Table II.)

these actions occurs, a clickstream event with timestamp, user, and position identification information is sent to the Content Database in the Backend.

Actions outside modes are also captured. An `enter` (`En`) / `exit` (`Ex`) event is created whenever a learner enters / exits a unit, as is a `login` / `logout` event whenever a learner logs in / out of the course. Also, each mode within a unit is contained in a separate window; learners can customize the window layout in their browser. A `window` event is created each time a learner maximizes (`Wx`) or minimizes (`Wn`) a window.

B. Dashboard: Instructor-facing

The Dashboard is divided into tabs, each with charts on a different learning aspect. The instructors for the courses considered in this paper had access to the three following tabs:
Overview. This provides summaries of learners' progress.
Engagement. Each learner is given an engagement score in each unit, in each module, and for the whole course. The computation of these scores will be discussed in Sec. III, as it is one of our machine learning features. This tab visualizes these scores for an instructor to draw comparisons.
Content. This shows time spent, number of views, and completion rate on each content mode, which will also be used as prediction features. A progress bar is shown for the average completion rate. Instructors can access plots of time spent and view count across each partition of a mode.

C. Discussion Forum

Our system integrates with NodeBB, an open-source discussion forum platform. Each course's forum is divided into threads, the first post in each thread made by the instructor.
User functions and data export. Within a thread, a user can create a `post` (consisting of some text), `reply` to a post, and `up-vote` or `down-vote` a post. At the end of a course, the NodeBB API provides the details of each thread to the SLN Database in the Backend. For each post, it indicates the user ID, timestamp, text, net votes (up-votes − down-votes), replies, and whether each reply was an instructor or a learner.

The interaction between learners in the discussion forum is an important part of the SLN. In Fig. 3, we illustrate interaction graphs for three course sessions considered in this paper (see Sec. III). Each node is a learner, and the weight $w_{i,j}$ from learner i to j is proportional to the number of times i posted and responded to j. We see that the structure in these short-courses is rather dense (with $\geq 34\%$ of the links non-zero, including learners who do not post that are not depicted in the graph) in contrast to the case of MOOCs [8]. This foreshadows an observation we will make in Sec. IV that differences in outcomes are more readily detected from the topical rather than the structural aspects of the discussions.

Table I summarizes the main event types from the Player and the Forum considered in this paper.

III. ML Features and Datasets

In this section, we present our behavior-based machine learning features. We will first specify the feature matrix that we compute for each dataset (Sec. III-A), then give descriptive statistics of datasets in terms of these features (Sec. III-B), and finally describe the feature selection process (Sec. III-C).

A. Our Machine Learning Features

Let $\mathbf{A} = [a_{v,f}]$ be the learner-feature matrix for a course, where $a_{v,f}$ is the value that feature $f \in \mathcal{F}$ takes for each learner v. We write $\mathbf{A} = [\mathbf{A}_c\ \mathbf{A}_s]$, where $\mathbf{A}_c$ and $\mathbf{A}_s$ are the matrices of content features and SLN features, respectively. In what follows, we define the quantities that comprise the corresponding feature subsets $\mathcal{F}_c$ and $\mathcal{F}_s$.

1) Content Features ($\mathcal{F}_c$): $\mathcal{F}_c$ summarizes the interactions a learner has with the Player. Event interactions consist of the six different types summarized in Table I: each of these types appears in $\mathcal{F}_c$ one time for each mode that they apply to. We use the frequency of events rather than indicator variables to account for how often learners use different behaviors. Additionally, $\mathcal{F}_c$ includes more summative quantities given in the Dashboard – time spent, completion rate, and engagement:
Time Spent. This is the amount of (real) time that a learner spent on each content mode. To compute the time spent on by a learner on a particular mode, we use a learner's clickstream events generated on that mode to reconstruct her behavior. In doing so, we account for cases in which a learner is obviously engaged in off-task behavior, e.g., if the duration between two events is extremely long, as in [1].
Completion Rate. This is the fraction of a mode that the learner completed, *i.e.,* the percentage of the content in that file that the learner visited as in [6].
Engagement. Engagement appears in $\mathcal{F}_c$ once per content mode (*i.e.,* file), once per unit, once per module, and once more as overall for the course.

Event	Description	Mode(s)
`play` (`Pl`)	A `play` event begins when a click event changes a learning mode to the `playing` state.	Slides
`pause` (`Pa`)	A `pause` is recorded when a click event changes a learning mode to the `paused` state.	Slides
`skip` (`Sb`, `Sf`)	A `skip back` (`forward`) occurs when a scrubber is brought to an earlier (later) position.	Slides
`scroll` (`Su`, `Sd`)	A `scroll up` (`down`) occurs when a scroll bar is brought to an earlier (later) position.	PDF, Article
`window` (`Wx`, `Wn`)	A `window max` (`min`) event occurs when a learning mode is maximized (minimized).	PDF, Article, Slides
`enter` (`En`, `Ex`)	An `enter` (`exit`) event occurs when a a learner enters (exits) a unit in the Player.	–
`post`	A `post` event happens when a user creates a post in a thread.	Forum
`reply`	A `reply` event occurs when a user creates a reply to a post.	Forum
`vote`	An `up-vote` (`down-vote`) occurs when a post receives an up-vote (down-vote).	Forum

TABLE I: Summary of the behavioral events analyzed in this paper, captured by the Player (content events) and Forum (SLN events).

File-level: Let $r_{v,o} \in [0,1]$ be the completion rate of user v on file o. Each file is further divided into a set of smaller partitions $\mathcal{P}(o)$, where $p \in \mathcal{P}(o)$ refers to the pth partition. For article and PDF, $\mathcal{P}(o)$ is the set of pages, and for interactive slides, $\mathcal{P}(o)$ is the set of one-minute video segments making up the full set of slides. Let $t_{v,p}$ be the time spent by user v on p, and let $\bar{t}_p$ be the "expected" time spent on p for normalization (defined below). Engagement on o is defined as:

$$e_{v,o}(r,t) = \min\left(\gamma \times r_{v,o} \times \prod_{p\in\mathcal{P}(o)} \left(\frac{1+t_{v,p}/\bar{t_p}}{2}\right)^{\alpha_t}, 1\right) \quad (1)$$

Here, $\alpha_t \geq 0$ is a parameter that models the diminishing returns property of the time spent component. Through this, a learner's time spent on each specific p counts incrementally less towards her engagement, *i.e.,* a learner is rewarded more for distributing her time spent across more partitions. The division by 2 makes the computation for each partition relative to a learner that registers the expected $t_{v,p} = \bar{t}_p$. $\gamma \in (0,1]$ is an instructor-specified constant that controls the spread of the overall engagement distribution; note that if completion $r_{v,o} = 1$ and the learner spends $t_{v,p} = \bar{t}_p$ on each p, then $e_{v,o} = \gamma$. We discuss the selection of γ and α_t in Sec. III-B.

Unit, module, and course-level: A weighted average is taken across the modes $\mathcal{O}(u)$ in a unit u to come up with the unit-level engagement: $e_{v,u} = \sum_{o\in\mathcal{O}(u)} \bar{t}_o e_{v,o} / \sum_o \bar{t}_o$ from (1) for each learner, where $\bar{t}_o$ is the expected length of o (defined below). In a similar manner, a weighted average is taken across units to come up with module and course-level engagements.

Normalization values: To calculate $\bar{t}_p$ and $\bar{t}_o$ for PDF and article, we first apply Optical Character Recognition (OCR) to obtain transcripts of the text, and manually correct any inconsistencies in the output. The reference time spent $\bar{t}_p$ on p is the expected time a learner will take to read the transcript of this partition, assuming an average reading speed of 6.6 characters per second. $\bar{t}_o$ is then $\sum_p \bar{t}_p$. For slides, $\bar{t}_p = 60$ sec $\forall p$, and $\bar{t}_o = 60|\mathcal{P}(o)|$ sec is the total length of the videos that comprise the interactive presentation.

2) SLN Features ($\mathcal{F}_s$): $\mathcal{F}_s$ contains quantities that summarize a learner's interaction within the SLN. This includes the frequency of the Forum events from Table II: the number of `posts` (and replies) a learner made, the number of `replies` the learner received, and the net `votes` the learner received on her posts/replies. It also includes the total number of words contained in said posts/replies. Finally, it includes the time period that a leaner stayed active in the forum, which we define as the time elapsed between the learner's first and last post.

Content similarity. $\mathcal{F}_s$ also contains features describing the contextual/topical aspect of a learner's posts. To measure the relevance of a learner's discussion to the course content, we define a content similarity measure $s_{v,u}$ between unit $u \in \mathcal{U}$ and learner $v \in \mathcal{V}$. The $s_{v,u}$ are included as features in $\mathcal{F}_s$ for each course unit. They are obtained as follows:

Topic distributions: We first extract the set of topics $\mathcal{K}$ in the course, and represent u's content and v's posts as probability distributions $\mathbf{d}_i = (d_{i,1}, ..., d_{i,|\mathcal{K}|})$ over the topics, where $i \in \mathcal{I} = \{1, ..., |\mathcal{U}| + |\mathcal{V}|\}$ indexes unit $u(i) = i$ if $i \leq |\mathcal{U}|$ and learner $v(i) = i - |\mathcal{U}|$ otherwise. To do this, we represent each i as a word frequency vector $\mathbf{w}_i = (w_{i,1}, ..., w_{i,|\mathcal{X}|})$ over the full dictionary $\mathcal{X}$ of words. For $i > |\mathcal{U}|$, $w_{i,x}$ is the number of times learner $v(i)$ wrote the xth word in $\mathcal{X}$ across all her posts, and otherwise $w_{i,x}$ is how many times the xth word appears in the text transcripts of $u(i)$.[1] In collecting words for $\mathcal{X}$ across the posts and content, we also apply appropriate stopword filtering, as in [8]. Then, with $\mathbf{W} = [\mathbf{w}_i] \in \mathbb{Z}^{|\mathcal{I}|\times|\mathcal{K}|}$ as the document-word matrix, we apply the Latent Dirichlet Allocation topic model [8], which results in the $\mathbf{d}_i$.

Similarity measure: With the topic distributions in hand, we define the similarity via total variation distance: $s_{v,u} = 1 - 0.5\|\mathbf{d}_{i(v)} - \mathbf{d}_{i(u)}\|_1$.[2] In this way, $s_{v,u} \in [0,1]$ captures the variation between the two topic-word distributions.

3) Time-varying features: For each course, we define $\mathbf{A}(n)$, and its subsets $\mathbf{A}_c(n)$ and $\mathbf{A}_s(n)$, to be the feature matrices using the behavior available from the launch of the course through day n. Evaluating using day-by-day data allows us to assess how the quality of our predictions is expected to vary at different points along the course timelines. Note that prior works on student performance prediction [1], [6] have used the equivalent of a unit-by-unit approach for early detection (*i.e.,* using data collected in the first few units). The day-by-day approach allows us to account for the fact that learners tend to re-visit units at different times throughout a course.

B. Datasets and Computed Features

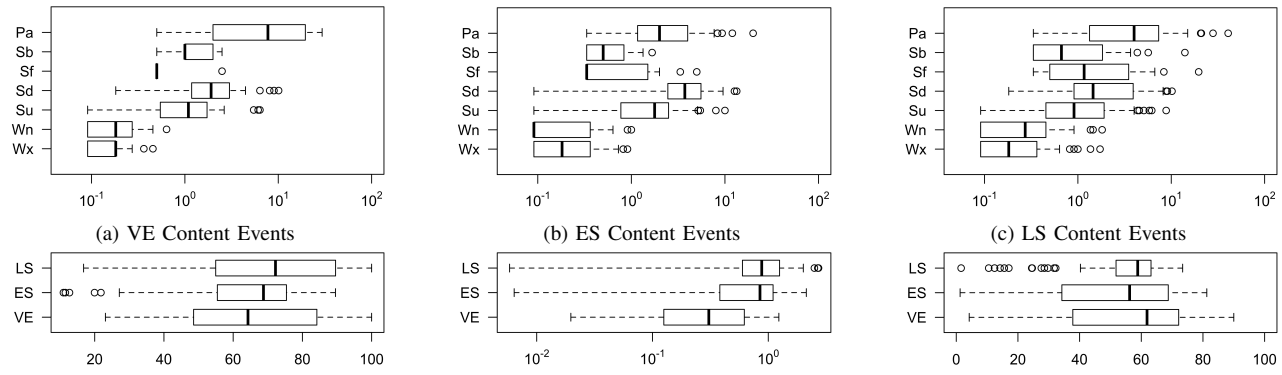
1) Courses and Datasets: The datasets used to evaluate our method are from three short-courses we hosted for a corporate

[1]Since text transcripts are for PDF and article modes only, this does not explicitly include the slides in a unit. However, for the courses we consider, we notice that the text is usually a repetition of the slide content.

[2]$i(u)$ maps from u to its index in $\mathcal{I}$, and likewise for $i(v)$.

Course Name		Days	Units	Slideshows	Articles	PDFs	Enrolled	Pass	Fail	Click	Post
Vanquishing Toxic Employees	VE	14	11	1	6	6	79	15	64	20,126	73
Effective Communication Skills	ES	14	11	2	4	8	94	45	49	45,380	104
Developing Leadership Styles	LS	14	11	2	6	5	96	44	52	48,449	116

TABLE II: Summary information of the short-course datasets used in this paper.



(a) VE Content Events (b) ES Content Events (c) LS Content Events

(d) Course-level engagement (e) Total time spent (hr) (f) Average completion rate (%)

Fig. 4: Boxplots of select content features $f \in \mathcal{F}_c$ computed for each dataset. (a)-(c) are for the event features, while (d)-(f) are for the analytic features that appear on the Dashboard.

training provider: "Vanquish Toxic Employees" (VE), "Effective Communication Skills" (ES), and "Techniques for Developing Your Leadership Styles" (LS). As the titles suggest, they emphasize business operations and leadership.

Learning outcomes: At the end of a course, each learner is given a single grade (pass, fail, extend, or expired). This outcome is assigned manually by the instructor as an unspecified combination of the learner's activity and participation levels throughout the course, using information from the Forum and Dashboard (see Sec. II). In our analysis, we group fail, extend, and expired into a single group (denoted fail), because the instructors view all three as undesirable.

Dataset summary: Summary information on these courses is given in Table II. They are roughly equivalent in their durations (14 days) and lengths (11 units). The courses contain between 20K and 50K clickstream events each, making the number of events generated by each learner in the Player rather large on average. The number of each type of mode (interactive slideshow, article, and PDF) in each course is also given here. LS and ES are well balanced in their ratio of Pass to Fail, but in VE, most of the learners (81%) fail.

Live events: The courses include 2-3 live sessions with instructors that are facilitated through the Forum. The first event is typically held one week in, aiming to "exchange thoughts and learning experiences." In Sec. IV-C, we will see that effective outcome predictions can be made starting around this time.

2) Statistics of Content Features: Fig. 4 gives distributions of several of the features in $\mathcal{F}_c$ for each dataset.[3] Each point in each plot corresponds to one learner. The events in (a)-(c) are aggregated over all modes in the course, and then normalized by the number of units in which that event can occur, for comparative purposes. The distributions in (d)-(f) are of course-level engagement, time spent across all file-level modes, and average file-level completion rate, respectively.

In comparing the distributions, we employ a Wilcoxon Rank Sum test for the null hypothesis that there was no difference between the distributions overall, and consider the p-values (p) from those tests.[4] We present the main and significant observations here (for more, see our technical report [12]):

(i) `Pa` *is most common*: This is especially true in VE, where the median (med.) number of pauses per unit is 9, and the effect is significant in comparing to other events ($p \leq$ 1E-3).

(ii) `Sd` *occurs more often than* `Su`: The shift is significant in ES and VE (med. from 1.1 to 1.7 and 1.9 to 3.7, $p \leq$ 7E-15).

(iii) VE has lower time spent: The time spent for VE compared to the other courses is shifted to the left (med. in VE $= 0.32$ hr vs. ≈ 0.85 in ES and LS, $p \leq$ 2.7E-11), even though the courses are each roughly the same length. This could be a reason for the outcomes being skewed towards fail in VE.

(iv) Engagement distributions are useful: We set $\gamma = 1$ and $\alpha_t = 0.1$ in (1) to generate engagement distributions with large ranges and relatively uniform spreads across the ranges. With this setting in each course, learner engagement varies from low values (≤ 23) to 100, with medians between 60 and 70, which makes it a useful metric for instructors to compare learners.[5] The fact that engagement is one of the most correlated content features for prediction in Sec. III-C also validates this choice.

3) Statistics of SLN Features: Fig. 5 gives the distributions of the following features in $\mathcal{F}_s$: word count in a learner's posts, word count in replies to a learner's posts, and posting time spread. Table III summarizes the five topics with highest support extracted from the posts and text content in each course. We make a few observations:

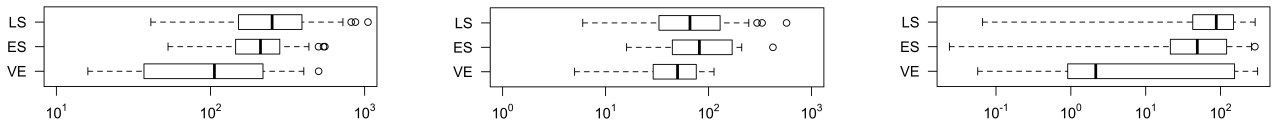
(i) SLN activity is significantly lower in VE: Each of the three features (posts, replies, and time spread) are lower in VE than in other courses (though only significant for word count, $p <$ 7.3E-5). This foreshadows a point we will see in Sec. III-C that SLN features are correlated with outcomes.

(ii) Topic words are relevant and supports are consistent:

[3] We only consider the non-zero values in these plots.

[4] Shapiro-Wilk tests detected significant departures from normality [6].

[5] VE is approximately normally distributed, with a Shapiro-Wilk $p > 0.03$.



(a) Post Word Count (b) Reply Word Count (c) Time Spread (hr)

Fig. 5: Boxplots of the main SLN features $f \in \mathcal{F}_s$ collected for each dataset.

VE			LS			ES		
k	$z_k(\%)$	top three words	k	$z_k(\%)$	top three words	k	$z_k(\%)$	top three words
1	12.7	employee problem toxic	1	14.9	leadership style kill	1	13.4	communicate email consider
2	12.5	jacki liza work	2	13.5	people style work	2	13.2	communicate skill effect
3	11.8	situation always difficult	3	13.5	team motivates focus	3	11.4	inform question feedback
4	9.6	conversation team behavior	4	12.3	time award emotion	4	10.6	person word language
5	9.5	behavior step toxic	5	10.7	high place change	5	9.6	listen understand paraphrase

TABLE III: Summary of the topics extracted with $|\mathcal{K}| = 5$. Given for each topic k are its support z_k and highest three constituent words.

From the titles of the courses, we see that the topics are representative of likely discussions for each course (*e.g.,* $k = 1$ in VE is about "toxic employees"), and are reasonably non-overlapping in the top words they include.

C. Feature Selection

The full feature matrix $\mathbf{A}(n)$ for each course has between 140 and 190 columns. In order to reduce overfitting and improve model interpretability, we perform feature selection prior to training the predictors on each $\mathbf{A}(n)$, $\mathbf{A}_s(n)$, and $\mathbf{A}_c(n)$. We implemented three standard methods: correlation analysis, information gain, and random forest importance [13].

Comparing the features selected from these methods in terms of their eventual predictive quality, we found that those selected by correlation analysis tended to yield the best results. In running correlation analysis on $\mathbf{A}(15)$ (*i.e.,* the full feature matrix built from all the course data), the selected behavioral features for each course are summarized in Table IV. We choose the top ten because prediction quality saturates beyond this point (for an analysis on the effect of varying the number of features, see Sec. IV.E of our technical report [12]). Noting that each of these ten have *positive* correlations with the course outcome, we make a few observations:

(i) Of the content features, the Dashboard quantities are more correlated: Engagement, time spent, and completion rate are more correlated with the outcome than the events in $\mathcal{F}_c$. With the exception of `enter`, events do not appear in the top-10.

(ii) The SLN features are more correlated than the content features: Features in $\mathcal{F}_s$ are more frequent in these lists than those in $\mathcal{F}_c$. The discussion post similarity features $s_{v,u}$ are notably important. The more relevant a learner's posts, the more familiar the learner is with the course content, which the instructors are likely to pick up on.

(iii) Word count is a correlated feature in all courses: A higher word count for a learner tends to imply a higher probability of successfully passing the course. Given that this feature is independent of any course content and/or structure, it may be useful for course-independent prediction algorithms.

IV. Prediction and Analytics

We now apply the feature sets from Sec. III to prediction. We first describe the algorithms and procedures used for evaluation (Sec. IV-A), then present and discuss our results (Sec. IV-B&C), and finally show examples of behavioral analytics that can assist instructors (Sec. IV-D).

A. Classifiers and Procedure

Prediction classifiers. We consider four classifiers for completeness: K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Linear Discriminant Analysis (LDA), and Random Forest (RF). We choose these for a few reasons. First, they have each demonstrated good performance in predicting student outcomes in other works, *e.g.,* KNN in [14], SVM in [1], [6], LDA in [15], [16], and RF in [17] (though only in [1], [6] with behavioral features). Second, given their optimization approaches, they are typically applied to different feature types, and we are using indicator, integer, and continuous features: for instance, RF is an ensemble tree method applied to any type of feature, whereas SVM uses a kernel function to find the optimal hyperplane separation and is typically applied to non-indicator features, and LDA has been seen to work better on continuous quantities given that it finds a linear combination of the features which best separates groups [16].

Parameters: For SVM, we use the radial basis function (rbf) kernel.[6] The parameters for SVM (kernel standard deviation (η) and regularization penalty (C)), RF (number of trees (τ) and number of variables (δ) randomly sampled at each tree split), and KNN (number of neighbors (κ)) are tuned during the cross validation procedure described below.

Metrics. We primarily consider AUC (*i.e.,* the area under the ROC curve) and Type II error (*i.e.,* fraction of fails that are incorrectly predicted as passes) as evaluation metrics. In practice, we are interested in identifying learners who are at risk of failing in advance so the instructor can be notified; consequently, we would like a classifier to obtain a low Type II error while maintaining a high AUC, so that we correctly identify the fails while not flagging too many passes incorrectly as fails. For completeness, we also report Accuracy (Acc, *i.e.,* fraction of all predictions that are correct).

Cross validation. For training and evaluation, we repeatedly (i) divide the dataset randomly into K folds ($K = 5$) stratified such that each fold has roughly the same proportion of passes

[6]We found this to obtain the best results out of all the standard kernels.

f	VE	ES	LS
1	Word Count	Post Similarity to Unit 2	Post Count
2	Post Similarity to Unit 5	Post Similarity to Unit 3	Post Similarity to Unit 3
3	Post Count	Post Similarity to Unit 8	Post Similarity to Unit 7
4	Time Spread	Unit 10 Article Engagement	Post Similarity to Unit 5
5	Session Count	Word Count	Post Similarity Between to Unit 2
6	Post Similarity to Unit 7	Unit 10 Engagement	Post Similarity to Unit 4
7	Post Similarity to Unit 6	Unit 11 Article Engagement	Word Count
8	Unit 5 Slideshow Completion	Unit 11 Engagement	Time Spread
9	Unit 5 Slideshow Engagement	Unit 10 `enter`	Unit 11 Article Engagement
10	Post Similarity to Unit 1	Unit 11 Article Completion Rate	Unit 11 Engagement

TABLE IV: List of the 10 behavioral features selected based on correlation analysis on the full matrix $\mathbf{A}(15)$ for each course. All of these features have positive correlations with outcome, and they are ordered from highest to lowest.

and fails, (ii) train and tune the algorithms through cross-validation on $K-1$ of the folds, choosing the set of parameters with highest average accuracy,[7] and (iii) evaluate on the holdout fold, similar to the procedure detailed in [1]. The metrics we report are averaged over several (50) runs of this procedure, to obtain a general estimate of quality.

B. End-of-Course Prediction

In Table V, we show the prediction results for each algorithm on the full feature set $\mathbf{A}(15)$, the SLN-only matrix $\mathbf{A}_s(15)$, and the content-only $\mathbf{A}_c(15)$ for each course. In practice, these are the results if the predictions are made once the courses are finished running. We make a few observations:

Behavioral data can be used for outcome prediction. Considering the full (combined) feature matrix $\mathbf{A}(15)$, we see that at least one of the algorithms is able to obtain a high quality prediction, which indicates that behavioral data can be used to make effective outcome predictions even when no assessment data is available. More specifically, for at least one of the algorithms, the AUC is larger than 0.82, while the Type II error is less than 0.11, meaning that less than 11% of the passes would be incorrectly identified as fails. RF, in particular, is able to obtain consistently high quality across each of the datasets (Acc > 0.81, AUC > 0.72, Type II < 0.18).

SLN features are more useful than content features by the end. Comparing the quality of predictions using SLN features ($\mathbf{A}_s(15)$) vs. content features ($\mathbf{A}_c(15)$) in Table V, we see that while the AUCs are roughly comparable across courses and algorithms (SLN being higher in $7/12$ cases), predictions on SLN features obtain substantially lower Type II errors (SLN is lower in all 12 cases). This implies that by the end of the course, classifiers using the SLN features are better able to avoid classifying those who fail as passing incorrectly.

Algorithm choice varies based on course and feature set. Considering the content features, SVM has poor quality across the three datasets (AUC ≤ 0.5 in two cases). Interestingly, this is in contrast to results in [1] which showed SVM to obtain high AUC (> 0.75) with similar features. In that application of predicting quiz performance in MOOCs, however, there are orders-of-magnitude more samples for training, and each learner appears in the dataset multiple times, which allows the SVM to include learner/quiz indicator features. These characteristics are not present in single-outcome courses. With SLN features, though, SVM's quality increases substantially.

C. Day-by-day Prediction

In Fig. 6, we evaluate the early detection capability of the full feature set for each course. To do this, we choose the algorithm with highest quality on $\mathbf{A}(15)$ for each course from Table V, and perform training and evaluation over $\mathbf{A}(n)$ for $n \in \{1, ..., 15\}$. In order to evaluate the effect that each group of features has over time, we repeat this over $\mathbf{A}_s(n)$ and $\mathbf{A}_c(n)$, and show the resulting AUC by day in Fig. 7. From these plots, we make a few observations:

Behavioral data has an early detection capability. In Fig. 6 we can see that, as expected, the quality of the predictors tends to rise from the beginning to the end of the course, with AUC and Acc increasing and Type II error decreasing. There is a tradeoff, then, between how early the predictions are applied and the expected quality. The following are two interesting points along the tradeoff in each course at which forecasts can be made in advance, each with reasonable quality:

(i) Detection midway through: The AUC hits a local maximum around the midpoint of the courses (day 6 or 7) – a trend which is more pronounced in VE and LS than in ES – hitting roughly 0.7 or higher in each case. This is right around the time of the first live event in the courses (see Sec. III-B1), which the instructors indicated is a useful point for the information to be provided. The Type II errors at these points are roughly 0.3.

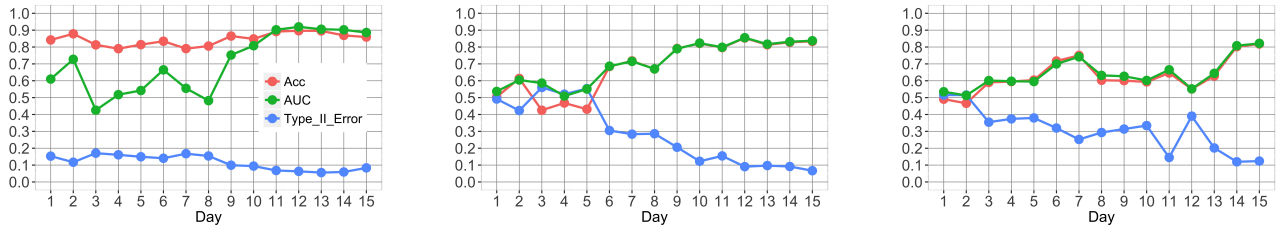
(ii) Detection three-quarters through: In VE and ES, the AUC saturates around three-fourths of the way through the course (day 10 or 11), at which point it is roughly 0.8 in both cases. The Type II errors have also dropped to roughly 0.1, meaning that we can expect 90% of fails to be correctly identified. If the final stretch of the course is sufficient time for instructor intervention, then this is a strong point to apply the algorithms.

For "earliest" detection, content features have an advantage. After the first half or so of each course in Fig. 7, we see that SLN features obtain higher AUC than content features, consistent with the observation in Sec. IV-B. For VE, this is true throughout the entire course. For ES and LS, however, the content features provide higher quality early, with a gain up to 0.1 in the first three days of VE. This indicates that content data may be more useful for detections that must be provided at the earliest stages of a course, consistent with an observation

[7]The set of parameters we test are $\eta \in \{0, 1, ..., 10\}$, $C \in \{$1E-5, 1E-4, ..., 1E5$\}$, $\kappa \in \{1, 2, ..., 10\}$, $\tau \in \{10, 11, ..., 300\}$, and $\delta \in \{1, 2, ..., 10\}$.

Course	Algo	Combined			SLN			Content		
		Accuracy	AUC	Type II	Accuracy	AUC	Type II	Accuracy	AUC	Type II
VE	RF	0.835 ± 0.011	0.727 ± 0.011	0.101 ± 0.011	0.857 ± 0.009	0.749 ± 0.009	0.093 ± 0.009	0.806 ± 0.010	0.594 ± 0.010	0.156 ± 0.010
	LDA	0.858 ± 0.012	$\mathbf{0.895 \pm 0.012}$	0.092 ± 0.012	0.752 ± 0.048	0.731 ± 0.048	0.061 ± 0.048	$\mathbf{0.830 \pm 0.012}$	$\mathbf{0.861 \pm 0.012}$	$\mathbf{0.102 \pm 0.012}$
	SVM	0.810 ± 0.001	0.500 ± 0.001	0.196 ± 0.001	0.804 ± 0.007	0.503 ± 0.007	0.186 ± 0.007	0.809 ± 0.001	0.500 ± 0.001	0.191 ± 0.001
	KNN	$\mathbf{0.865 \pm 0.010}$	0.796 ± 0.010	$\mathbf{0.068 \pm 0.010}$	$\mathbf{0.873 \pm 0.011}$	$\mathbf{0.786 \pm 0.011}$	$\mathbf{0.080 \pm 0.011}$	0.802 ± 0.010	0.630 ± 0.010	0.141 ± 0.010
ES	RF	0.827 ± 0.012	0.824 ± 0.012	0.179 ± 0.012	0.789 ± 0.007	0.790 ± 0.007	0.243 ± 0.007	$\mathbf{0.812 \pm 0.005}$	0.820 ± 0.005	$\mathbf{0.261 \pm 0.005}$
	LDA	0.750 ± 0	$\mathbf{0.843 \pm 0}$	0.25 ± 0	0.800 ± 0	$\mathbf{0.828 \pm 0}$	0.273 ± 0	0.800 ± 0	$\mathbf{0.869 \pm 0}$	0.273 ± 0
	SVM	$\mathbf{0.826 \pm 0.011}$	0.829 ± 0.011	$\mathbf{0.086 \pm 0.011}$	$\mathbf{0.816 \pm 0.012}$	0.821 ± 0.012	$\mathbf{0.087 \pm 0.012}$	0.630 ± 0.009	0.609 ± 0.009	0.280 ± 0.009
	KNN	0.817 ± 0.001	0.821 ± 0.008	0.222 ± 0.009	0.753 ± 0.002	0.746 ± 0.002	0.250 ± 0.002	0.741 ± 0.005	0.755 ± 0.005	0.345 ± 0.005
LS	RF	0.813 ± 0.012	0.808 ± 0.012	0.179 ± 0.012	$\mathbf{0.850 \pm 0.011}$	0.849 ± 0.011	0.130 ± 0.011	$\mathbf{0.722 \pm 0.013}$	$\mathbf{0.726 \pm 0.013}$	$\mathbf{0.204 \pm 0.013}$
	LDA	0.781 ± 0.012	0.851 ± 0.012	0.216 ± 0.012	0.785 ± 0.0137	$\mathbf{0.853 \pm 0.014}$	0.192 ± 0.014	0.679 ± 0.014	0.725 ± 0.014	0.266 ± 0.014
	SVM	0.817 ± 0.010	$\mathbf{0.822 \pm 0.010}$	$\mathbf{0.102 \pm 0.010}$	0.828 ± 0.012	0.831 ± 0.012	$\mathbf{0.113 \pm 0.012}$	0.515 ± 0.009	0.490 ± 0.009	0.468 ± 0.009
	KNN	$\mathbf{0.821 \pm 0.012}$	0.562 ± 0.012	0.190 ± 0.012	0.828 ± 0.010	0.503 ± 0.010	0.186 ± 0.010	0.644 ± 0.014	0.635 ± 0.014	0.302 ± 0.014

TABLE V: Prediction quality of the algorithms on the content, SLN, and combined feature sets at the end of the course ($\mathbf{A}_c(15)$, $\mathbf{A}_s(15)$, and $\mathbf{A}(15)$). For each metric, we report the average and standard deviation across 50 cross validation trials. The algorithm obtaining the best value for each course-feature-metric triple is bold. Overall, we see that behaviors can be used for quality outcome predictions.



(a) LDA applied to VE (b) SVM applied to ES (c) SVM applied to LS

Fig. 6: Variation in prediction quality by day for each course, using the full feature set. At day n, the predictor is using $\mathbf{A}(n)$ for training. The AUCs reach 70% by day 7, which shows that behavioral features can be used for early detection in short-courses.

in [1] for MOOCs. This phenomenon can be explained by the fact that a course's SLN develops and evolves over time: initially, the discussions are more small-talk in nature [3], at which point learners are *e.g.,* introducing themselves and learning the content individually. After this initial phase passes (*e.g.,* around the first live event), the network has begun to mature, and becomes increasingly important to learner success.

D. Feature Correlation Analysis

We also analyze how the correlations of the top features vary over time. In practice, this can give instructors insight into which specific behaviors are most related to eventual learning outcomes at different points. Combining this with the list of learners predicted to fail can lead to recommendations on how those learners can improve their chance of success.

Referring to the features in Table IV, the plots of the top 5 are given in Fig. 8. We make a few observations here (for more, see our technical report [12]):

Rank convergence. The feature correlations generally become stronger over time with more data, as expected. The values converge to between 0.5 and 0.75 around days 10-11, which is consistent with the saturation of prediction quality in Fig. 6. The increases are not monotonic, however: there are points, particularly in the first week for VE and ES, where the correlations drop. These are times where learners who end up failing are participating in the discussions, so the instructors can attempt to engage the learners posting in these periods.

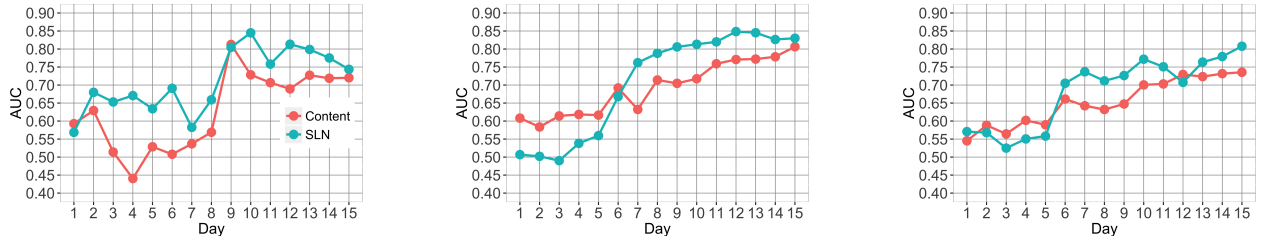
Content discussion recommendations. As discussed in Sec. III-C, the top features for each course include discussion post similarity to specific units. Analyzing the trends of these correlations leads to some interesting findings that can be turned into SLN discussion recommendations. In LS, notice that "post similarity to unit 7" has remarkably low correlation compared with the other features until day 9, even though in the end it is the third most correlated. This is likely because this unit is far down the syllabus, so learners are not focusing on this content until later, and therefore it may be beneficial to give advanced warning on the importance of this content. In ES, the correlation of "similarity to unit 3" also kicks in at a much slower rate than we would expect, given that "similarity to unit 2" is from a neighboring unit and persists quickly.

V. Related Work

Researchers have developed predictive learning analytics to forecast different attributes of students in advance, *e.g.,* how they will perform on assessments [1], [6], their risk of failing [18], their final grades [7], [10], whether they will drop out [5], [10], and whether forum intervention will be needed [11]. Our work considers the unique case of learners in short-courses with no assessment data for modeling, making most of these not directly applicable. Instead, our method relies solely on content and Social Learning Network (SLN) behaviors.

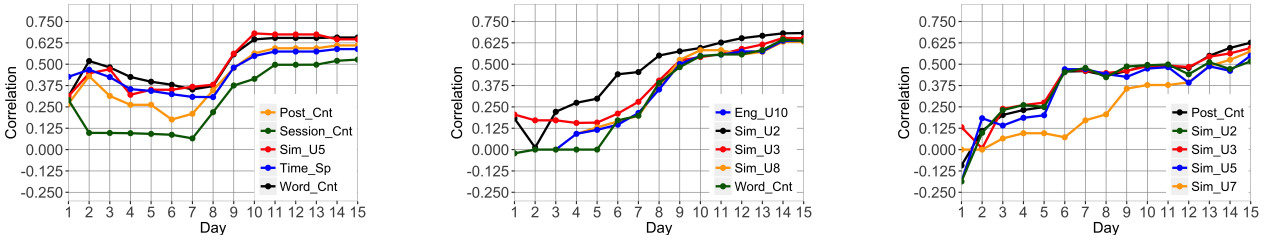
A few of these works [1], [5], [6] have used video-watching clickstream data as learning features in MOOC. In particular, [6] trained an SVM with a similar set of content features, which we saw obtained low quality in our scenarios. The content modes we consider here – interactive slideshows, articles, and PDFs – are common in online courses outside of MOOC, and our system enables collection of this data too. Our method is also based on day-by-day rather than *e.g.,* quiz-by-quiz prediction as in [6], which is more practical for early detection in these short-courses where learners re-visit content.

Regarding SLN more generally, several studies have emerged recently on *e.g.,* MOOCs [8], [10], [11], Q&A sites [3], and enterprise social networks [19]. Similar to [10], [11], we apply SLN features to prediction; [10] incorporates post and reply frequency into a probabilistic graphical model to predict grades and completion, and [11] predicts whether instructor participation in threads will be needed from semantics and explicit references to course files. Different from these



(a) VE (b) ES (c) LS

Fig. 7: Variation in prediction quality by day for each course, using the content ($\mathbf{A}_c(n)$) and SLN ($\mathbf{A}_s(n)$) features. In ES and LS, the SLN features have higher quality beyond the first few days, while the content features are useful for earliest detection.



(a) VE Features Correlation (b) ES Features Correlation (c) LS Features Correlation

Fig. 8: Variation in feature correlation by day for the top 5 correlated features for each course, corresponding to Table IV ("Sim_UX" is "post similarity to unit X"). Correlation gradually increases through days $10-11$ in each case, where it stabilizes.

works, in addition to structural SLN attributes our method incorporates topic similarity between posts and content, which we find is particularly predictive in our short-course scenarios.

VI. Conclusion and Future Work

We presented a method for predicting learning outcomes from learner behavior in online short-courses. The lack of intermediate assessments in this type of course makes the development of predictive learning analytics particularly challenging. Our method relies solely on behavior-based machine learning features, including a learner's interaction with the content integrated into a course and with one another in Social Learning Networks (SLN). Evaluating on data collected from three short-courses hosted through our system, we obtained high prediction quality by the middle stages of the courses, underscoring the capability of our method to provide early detection to instructors. We also observed that SLN attributes became the more useful set of behaviors for prediction over time, while the content attributes provided better quality for "earliest" detection in the first few days. Further, we found that our method can generate behavioral analytics.

In the future, we plan to investigate other content and SLN features, as well as other forms of classifiers that may enhance prediction quality further. We will also incorporate the methods described here into our Dashboard, so that instructors of these short-courses can access the predictions in an online manner during future course sessions. This will allow us to collect feedback on our method, and to measure changes in pass rates resulting from interventions made based on the predictions and analytics – the ultimate measure of efficacy.

Acknowledgment

We thank the rest of our team at Zoomi, and the reviewers for their valuable comments. This work was in part supported by NSF CNS-1347234 and ARO W911 NF-14-1-0190.

References

[1] C. G. Brinton, S. Buccapatnam, M. Chiang, and H. V. Poor, "Mining MOOC Clickstreams: Video-Watching Behavior vs. In-Video Quiz Performance," *IEEE Trans. Sig. Proc.*, vol. 64, no. 14, pp. 3677–3692, 2016.

[2] S. Kimmel. (2015) Training Evolution: The Current and Future State of Corporate Learning Modalities. http://www.workforce.com/articles.

[3] C. G. Brinton and M. Chiang, "Social Learning Networks: A Brief Survey," in *IEEE CISS*, 2014, pp. 1–6.

[4] J. Bersin. (2016) A Bold New World of Talent, Learning, Leadership, and HR Technology Ahead. http://marketing.bersin.com.

[5] T. Sinha, P. Jermann, N. Li, and P. Dillenbourg, "Your Click Decides Your Fate," in *ACL EMNLP*, 2014, pp. 3–14.

[6] C. G. Brinton and M. Chiang, "MOOC Performance Prediction via Clickstream Data and Social Learning Networks," in *IEEE INFOCOM*, 2015, pp. 2299–2307.

[7] Y. Meier, J. Xu, O. Atan, and M. van der Schaar, "Predicting Grades," *IEEE Trans. Signal Proc.*, vol. 64, no. 4, pp. 959–972, 2016.

[8] C. G. Brinton, S. Buccapatnam, F. M. F. Wong, M. Chiang, and H. V. Poor, "Social Learning Networks: Efficiency Optimization for MOOC Forums," in *IEEE INFOCOM*, 2016.

[9] GP and Training Industry. (2010) Delivering Virtual Instructor-Led Training. http://www.salt.org/.

[10] J. Qiu, J. Tang, T. X. Liu, J. Gong, C. Zhang, Q. Zhang, and Y. Xue, "Modeling and Predicting Learning Behavior in MOOCs," in *ACM WSDM*, 2016, pp. 93–102.

[11] M. Kumar, M.-Y. Kan, B. C. Tan, and K. Ragupathi, "Learning Instructor Intervention from MOOC Forums: Early Results and Issues," *ERIC EDM*, pp. 218–225, 2015.

[12] Technical report. http://cbrinton.net/EDOSC-tech.pdf.

[13] Y. Chen and C. Lin, "Combining SVM with Various Feature Selection Strategies," in *Feature Extraction*. Springer, 2006, pp. 315–324.

[14] A. Toscher and M. Jahrer, "Collaborative Filtering Applied to Educational Data Mining," *KDD Cup*, 2010.

[15] L. V. Morris, S.-S. Wu, and C. L. Finnegan, "Predicting Retention in Online General Education Courses," *American Journal of Distance Education*, vol. 19, no. 1, pp. 23–36, 2005.

[16] S. Guo and W. Wu, "Modeling Student Learning Outcomes in MOOCs."

[17] Z. A. Pardos and N. T. Heffernan, "Using HMMs and Bagged Decision Trees to Leverage Rich Features of User and Skill from an Intelligent Tutoring System Dataset," *Journal of Machine Learning Research*, 2011.

[18] H. Lakkaraju, E. Aguiar, C. Shan, D. Miller, N. Bhanpuri, R. Ghani, and K. Addison, "Machine Learning Framework to Identify Students at Risk of Adverse Academic Outcomes," in *ACM KDD*, 2015, pp. 1909–1918.

[19] J. Cao, H. Gao, L. E. Li, and B. Friedman, "Enterprise Social Network Analysis and Modeling: A Tale of Two Graphs," in *IEEE INFOCOM*, 2013, pp. 2382–2390.