Learning about Social Learning in MOOCs: From Statistical Analysis to Generative Model

Christopher G. Brinton, *Student Member, IEEE*, Mung Chiang, *Fellow, IEEE*, Shaili Jain, Henry Lam, Zhenming Liu, and Felix Ming Fai Wong

Abstract—We study user behavior in the courses offered by a major massive online open course (MOOC) provider during the summer of 2013. Since social learning is a key element of scalable education on MOOC and is done via online discussion forums, our main focus is on understanding forum activities. Two salient features of these activities drive our research: (1) *high decline rate*: for each course studied, the volume of discussion declined continuously throughout the duration of the course; (2) *high-volume, noisy discussions*: at least 30 percent of the courses produced new threads at rates that are infeasible for students or teaching staff to read through. Further, a substantial portion of these discussions are not directly course-related. In our analysis, we investigate factors that are associated with the decline of activity on MOOC forums, and we find effective strategies to classify threads and rank their relevance. Specifically, we first use linear regression models to analyze the forum activity count data over time, and make a number of observations; for instance, the teaching staff's active participation in the discussions is correlated with an increase in the discussion volume but does not slow down the decline rate. We then propose a unified generative model for the discussion threads, which allows us both to choose efficient thread classifiers and to design an effective algorithm for ranking thread relevance. Further, our algorithm is compared against two baselines using human evaluation from Amazon Mechanical Turk.

Index Terms—MOOC, social learning networks, data mining, regression, concept learning

1 Introduction

The recent and rapid development of massive online open courses (MOOCs) offered through platforms such as Coursera, edX, and Udacity demonstrates the potential of using the Internet to scale higher education. Aside from business models and potential impact, pedagogy is an often–debated subject as MOOCs try to make higher education available to a broader base. Low completion rates have often been cited to highlight a scale–efficacy tradeoff [1], [2], [3], [4].

Social learning is a key aspect of MOOC platforms. It holds the promise of scalable peer-based learning and is often the dominant channel through which teachers and students can interact. As these platforms proliferate, a natural question arises: How can we leverage the large-scale, extensive data that has emerged in recent months to better understand MOOC forum activities?

It has been observed that these forums suffer from the following major problems [5], [6]:

- *Sharp decline rate*. The amount of interaction rapidly drops soon after a course is launched.
- *Information overload* [7]. As a course reaches a larger audience, its forum is often flooded by discussions from many students. Thus, it quickly becomes infeasible for anyone to navigate the discussions to find course-relevant information.

In this paper, we study both problems through a comprehensive data set that we obtained by crawling the discussion forums of all courses that were offered on Coursera during the summer of 2013. Through examination, we quickly discovered that both of these problems are ubiquitous on Coursera (see Figs. 1 and 2). Thus, we believe it is natural to ask the following two questions:

Question 1 (Q1). How rapidly does the participation rate in the forums decline over time, and what behavioral factors maintain a healthy participation rate?

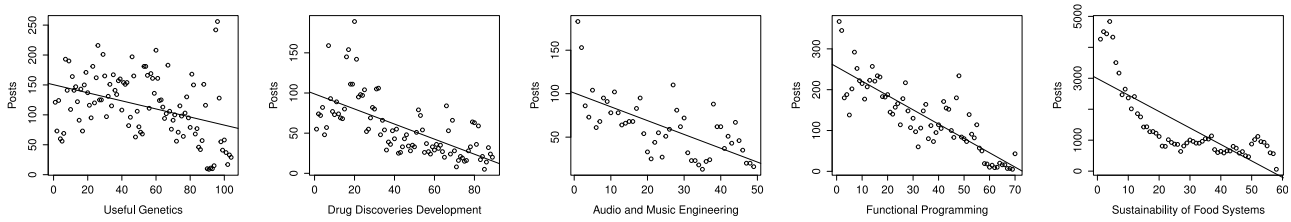
Question 2 (Q2). Is there a way to codify the generative process of forum discussions into a simple model, and if so, can we leverage such a model to facilitate user navigation?

Motivation. The motivation behind studying Q1 is as follows. Collaborative learning is an essential component of the educational experience for many students in online courses [8]. Since the discussion forums are the main venue for teacher-to-student interaction and the only venue for student-to-student peer learning on MOOC, understanding these dynamics of these forums is an important part of assessing the quality of student learning on these platforms. Note, however, that not all students choose to participate on these forums; some lurkers will follow the material in a course without socializing [9]. The question of how to encourage lurker students to participate, and even whether this would be beneficial to them individually in the first place, is beyond our scope here.

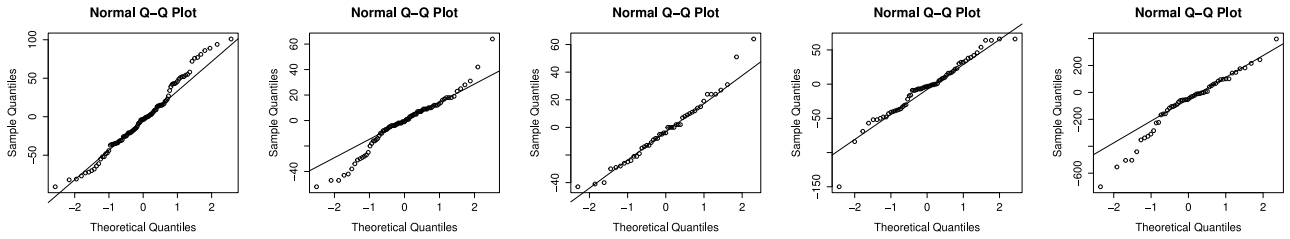
As for Q2, crystalizing the formation of discussion threads into a simple model will allow us to address the information overload problem and in turn improve the online learning experience. Addressing information overload problems traditionally falls into the area of information

- *C.G. Brinton, M. Chiang, Z. Liu, and F.M.F. Wong are with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08540. E-mail: {cbrinton, chiangm, zhenming, mwthree}@princeton.edu.*
- *S. Jain is with Microsoft, Bellevue, WA 98004. E-mail: shj@microsoft.com.*
- *H. Lam is with the Department of Mathematics and Statistics at Boston University, Boston, MA 02215. E-mail: khlam@bu.edu.*

Manuscript received 28 Dec. 2013; revised 23 May 2014; accepted 29 June 2014. Date of publication 9 July 2014; date of current version 15 Dec. 2014.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TLT.2014.2337900



(a) Number of posts per day



(b) Difference in post counts between two consecutive days

Fig. 1. The decline of forum activities over time. (a): We randomly choose five of the 73 courses we crawled and plot the number of posts over the days. A regression line with regressor on time is added. (b): Q-Q plots for these courses of the difference in count between two consecutive days, after removing 6 percent outliers (see Section 3).

retrieval (IR)[1] [10]. The primary goal here, however, is to highlight the unique characteristics of the dynamics of MOOC forums that are not seen in other online forums such as Yahoo! Q&A and Stackoverflow (or other social media sites such as Twitter and Facebook). The generative model we develop in Section 4 guides us in choosing classifiers to filter out "noise" in discussions and in designing ranking algorithms to find the most course-relevant discussions.

Our methodology. Our analysis consists of the following components.

(1a) Statistical analysis. To address Q1, we carry out an in-depth analysis to understand the factors that are associated with student participation on MOOC forums. We first use regression models to understand what variables are significantly correlated with the number of posts (or users) on the forums in each day for each course. As an example, one of the interesting discoveries is that higher teaching staff participation in the discussions is associated with a higher discussion volume, but it does not slow down the decline rate. We also apply a standard t-test procedure to understand whether the creation of too many new threads in a short time will reduce the depth of discussion in each thread. Along the way, we also present some basic statistics about Coursera, such as the total number of students that participate in the forums, the distribution on the number of posts for each student, and the distribution on thread lengths.

(1b) Identifying the information overload problem. Based on the statistical analysis, it is apparent that users have different needs in different stages of the course:

- In the first few days, the forum is often flooded with "small-talk," such as self-introductions. The primary goal in this stage is to *classify* these threads and filter them out.
- Beyond the first few days, the volume of small-talk often begins to drop. At this point, most of the threads are valuable, so it is important to be able to give a relevance ranking of new threads over time.

Thus, we need both an effective discussion-thread classifier and a relevance-ranking algorithm. But we want to understand a more fundamental question, proposed in Q2: is there a unified mechanism under which we can consider these designs? This is addressed next.

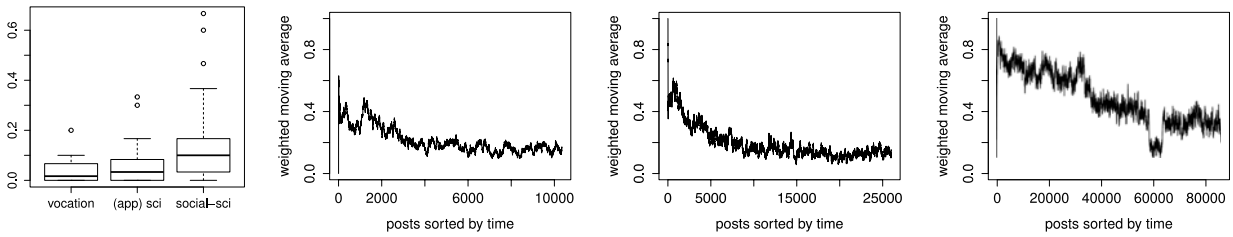
(2) Generative models. We propose a *unified generative model* for thread discussions that simultaneously guides (i) the choice of classifiers, (ii) the design of algorithms for extracting important topics in each forum, and (iii) the design of a relevance ranking algorithm based on the resulting topic extraction algorithm. We also compare our ranking algorithm to a number of baseline algorithms through *human evaluation*. Our simple model explains all the key experimental results we witness.

Related work. There has been a great deal of research in the areas of online social interaction and forum dynamics, as well as online education. Here, we highlight a number of recent, key works in these areas.

MOOCs. Piech et al. [11] designed algorithms for a peer grading system to scale up student evaluation in MOOCs. In our work, we are considering a different aspect of efficacy on these platforms: attempting to foster engagement in discussion forums. Along these lines, Ahn et al. [12] used a longitudinal fixed-effects model to identify the influencing factors of student participation in peer-to-peer learning for a MOOC-type platform. Kizilcec et al. [13] argued that student engagement in MOOCs should not be based solely on regular coursework completion. Cheng et al. [14] designed tools to predict a student's risk of dropping out of a course.

Compared to the above, our study is unique in that: (1) it is based on a much more comprehensive data set, 78 courses versus at most 7 in previous work; (2) it identifies new factors influencing engagement; and (3) it crystallizes discussion dynamics via a generative model.

1. Research in IR provides solutions to prioritize and personalize how information is presented to a user, through the application of techniques like automatic text categorization and link analysis.



(a) Sampled portion of small-talk by course category.

(b) Moving averages on the portion of small-talk by course category. Left: vocation; Mid: (appl) sci; Right: social-sci.

Fig. 2. Statistics of small-talk by course category. Categories are vocational (vocation), sciences and applied sciences ((appl) sci), and humanities and social sciences (social-sci).

Forums for online education. In the context of analyzing forum discussion for online education, Lui et al. [15] studied the feasibility in performing automated text categorization for a small online messaging board supporting a traditional classroom. This work focuses on one specific course and it is unclear how their techniques can be generalized. On the theoretical side, Ghosh and Kleinberg [16] took a game-theoretic approach to quantify the optimal rate of instructor participation to foster student discussion. See also [8] for a survey of earlier relevant works on this subject.

Online forums outside MOOC. There are two main lines of research on the dynamics of social forums: (1) understanding social interactions in forums (e.g., [17], [18], [19], [20]), and (2) finding high-quality information or users in the discussions, by applying link analysis techniques on user interaction graphs (e.g., [21], [22], [23]) or machine learning techniques on a combination of user and thread features (e.g., [24], [25], [26]).

Note that the discussion forums in MOOCs differ from other social media studied in the following ways. First, *both social and technical discussions are encouraged.* On the contrary for example, in Stackoverflow [25], administrators aggressively remove low-quality threads, and on Twitter [18], very few technical discussions occur. Second, *each forum focuses on one course.* Each course has one forum, and only students enrolled in the course can participate on it. A large portion of the students are also first-time users. While Yahoo! Q&A [17], [23] has both social and technical discussions, MOOC forums have weaker social interactions and more focused technical discussions (and thus the techniques developed in [17] are not directly applicable).

2 Preliminaries

This section presents how we collected our data set and gives an overview of the Coursera platform.

Collecting the data set. We focused on all 80 courses that were available in the middle of July and that ended before August 10, 2013. Seven of these 80 courses became inaccessible while we were crawling or coding the data, and thus the 73 courses for which we have complete records were used for statistical analysis. On the other hand, with the generative model it is less important to have a complete forum data set, and so we added five more courses that ended shortly after August 10. Table 1 gives the entire list of courses in our dataset, indicating those used in different sections.

Procedurally, we first manually calculated various properties for each course, such as the total video length and whether it was quantitative or vocational. Then, we crawled the forum content from Coursera's server at a rate of 1 to 3 pages per second using Python and the Selenium library. Finally, we used Beautifulsoup to parse the html into text files. In total, our data set consists of approximately 830K posts (Section 3 presents more details).

Categorizing the courses. For the purpose of comparison across course types, we categorize a course as *quantitative* or non-quantitative, and *vocational* or non-vocational. We adopt the following definitions:

- If a substantial portion of a course requires the students to carry out mathematical or statistical analysis, or to write computer programs, then it is quantitative.
- If a course's material is directly relevant to jobs that require high school or college degrees, or it is plausible to see the course offered in a typical university's continuing education division, then it is vocational.

Among the 73 courses, 37 of them were considered quantitative and eight of them vocational. There are six courses that were both quantitative and vocational. For summary purposes, we use these categories to partition the data into three groups: a course could be (1) vocational, (2) science or applied science (i.e., quantitative but not vocational), or (3) humanities and social sciences (i.e., neither quantitative nor vocational).

Forum structure. A MOOC forum consists of a number of threads. Students are able to create new threads or add content to existing threads. Each of these threads consists of one or more "posts," sorted in chronological order. The first post is written by the person who created the thread. A user may respond to a thread (i.e., add a new *post*) or respond to a post (i.e., add a new *comment*). In our analysis, we do not distinguish between posts and comments for the following two reasons: (1) There are only a small portion of comments in our data set; and (2) Due to the UI, a student may be unaware of whether she is making a comment or adding a post.

Forum topics. The discussion threads can be roughly categorized into the following groups:

- *Small-talk* conversations that are not course-related, such as a self-introductions or requests to form study groups.
- *Course logistics* such as when to submit homework, how to download lecture videos, and so on.
- *Course-specific discussions* that can range in scope from highly specific to open-ended.

The last two groups are sometimes referred to collectively as *course-related discussions*. A similar taxonomy distinguishing between conversational (i.e., small-talk) and informational (i.e., course-related) discussions is given in [24]. Note that both small-talk and course-related discussions are important to the forum experience on MOOC. At the same time, we can identify a number of scenarios in which a student would prefer to read course-related threads only, such as when she is reviewing for an exam. Hence, it is important to be able to separate conversational from informational discussions in MOOC, in order to assist user navigation as needed, especially when the number of new threads is already excessive.

As a result, we want to understand (1) how many small-talk posts exist for each course, and (2) whether or not the portion of small-talk changes over time.

We investigate the first question with the help of Amazon Mechanical Turk (MTurk)[2] [27]. Specifically, we randomly chose 30 threads from each course and hired workers from MTurk to label them. Each thread was labeled by three workers; we used a majority vote among them to assign the final labels.

Fig. 2a shows the distributions of small-talk by course category. We can see that a substantial portion of the course forums have more than 10 percent small-talk. Also, those in the humanities and social sciences category tend to have a higher portion of small-talk than the others.

Temporal dynamics of small-talk. We now study how the portion of small-talk changes over time. Since it is infeasible (in time and cost) to label a significant portion of the threads, we use a machine learning approach (specifically, a support vector machine, see Section 4) to classify the threads, using the labels from MTurk as training data. We put all threads under the same course category together. Then, we sort the threads by the time elapsed between the beginning of the course and the creation of the thread, focusing only on the threads created within 35 days of when the class started. Finally, we compute the "moving average" as follows: Let $h_1, h_2, \ldots, h_m$ be the sorted threads within the same group and η_i be an indicator variable that sets to 1 if h_i is classified as small-talk, and 0 otherwise. The moving average at time t is given by $s_t = \frac{\sum_{1 \le i \le t} \eta_i \alpha^{t-i}}{\sum_{1 \le i \le t} \alpha^i}$.

Fig. 2b shows the results for $\alpha = 0.99$. We can see that at the beginning, the percentage of small-talk is high across different categories, and then it drops over time. However, for humanities and social sciences courses, on average more than 30 percent of the threads are classified as small-talk even long after the start dates.

We remark that these plots only give estimates of the volume of small-talk. There are two types of noise present here. First, we are aggregating all threads in the same category together, so course-level information could be lost. Second, the support vector machine may have classification errors. Nevertheless, we may conclude that small-talk is a major source of information overload on MOOC forums.

Why the existing infrastructure is insufficient. We note that Coursera allows users to categorize threads. But these categories are customized by the teaching staff, and some categorizations are more effective than others. Further, there is no effective mechanism to force the students to abide by these definitions consistently. And obviously, it would be infeasible for the staff to manually correct labels when new threads flood into the forums.

3 Statistical Analysis

This section examines the following two questions of Q1: (1a) What factors are associated with the volume and decline rate of MOOC forum participation? (1b) Will having more discussion threads in a short period of time dilute student attention? We use linear regression to answer (1a) and student's t-test to answer (1b).

3.1 Analysis of Forum Activity Decline

Here, we will investigate the relationship between certain factors and the forum activity decline rate. Our dependent variables are $y_{i,t}$ and $z_{i,t}$, where $y_{i,t}$ refers to the number of *posts* on the tth day in the ith course, and $z_{i,t}$ refers to the number of *distinct users* that participate in discussion on the tth day in the ith course. Both of these are important to social learning in MOOCs.

The following factors could be relevant to $y_{i,t}$ and $z_{i,t}$:

Quantitative (Q_i). An indicator (boolean) variable that sets to 1 if and only if the ith course is quantitative.

Vocational (V_i). An indicator variable that sets to 1 if and only if the ith course is vocational.

Video length (L_i). The sum of the length of all lecture videos in the ith course (in hours).

Duration (D_i). The total length (in days) of the ith course.

Peer-grading (P_i). An indicator variable that sets to 1 if and only if at least one assignment in the ith course is reviewed/graded by peer students.

Staff activity (S_i). The number of posts the teaching staff makes throughout the ith course.

Graded homework (H_i). The total number of homework assignments that are graded by the teaching staff.

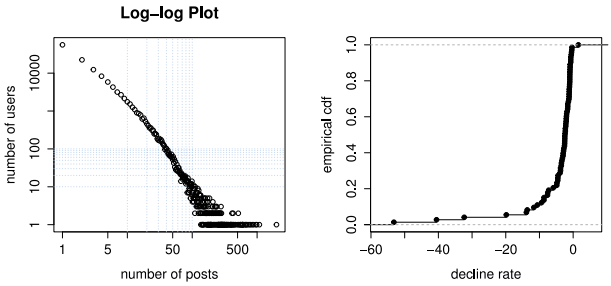
Intrinsic popularity (M_i or M_i'). The volume of forum discussion in the beginning of the course. If the dependent variable is the number of posts $y_{i,t}$, this is defined as M_i, the median number of posts in the first three days of the ith course; if it is the number of users $z_{i,t}$, then this is defined as M_i', the number of distinct users in the first three days. Roughly speaking, this variable captures the "intrinsic popularity" of each course, e.g., it is likely that a course on public speaking will be more popular than a typical course in electrical engineering.

We will describe the empirical behavior of these variables before we present and analyze our model.

3.1.1 Statistics of Coursera

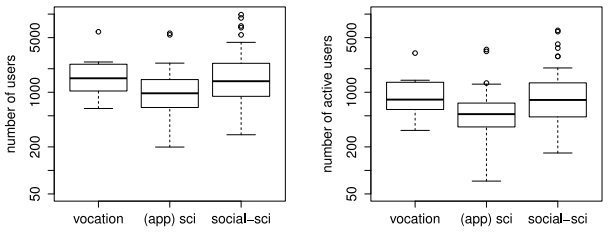
We now examine the key statistics of 73 courses (see Table 1), starting with student behavior.

2. MTurk is a crowdsourcing service provided by Amazon which allows anyone to design tasks that they can pay workers to solve online. We use MTurk in our work because of the large volume of data that needs to be labeled manually, and also to remove bias that could result from us generating these labels ourselves.



(a) Number of posts per student. (b) CDF of decline rate.

Fig. 3. Distribution of student posts and decline rates.



(a) Number of users. (b) Number of active users.

Fig. 4. Distribution of the number of students (left) and active students (right) per course, by category.

Students in the forums. There are 171,197 threads, 831,576 posts, and 115,922 distinct users in our data set. Fig. 3a shows the distribution of the number of posts each student made (in log-log scale).

We can roughly classify each student as an "active" or "inactive" forum user. We consider a student active if she made at least two posts in a course,[3] and inactive otherwise. Fig. 4 shows the distribution of the number of students and active students in different courses, separated by category. The reduction is substantial: while the average number of students per course is 1,835.0 (standard deviation (sd) $= 1{,}975.4$), the average for active students is only 1,069.7 (sd $= 1{,}217.7$). *Enrollment across courses.* We remark that 102,782 students (88.7 percent) were only enrolled in one course, while only 9,938 (8.6 percent) and 3,202 students were enrolled in two and more than two courses, respectively.

Distribution of decline rate. We now present how the number of posts $y_{i,t}$ and the number of users $z_{i,t}$ change over time for different courses. Fig. 1a shows the variables $\{y_{i,t}\}_{t\geq 1}$ for five randomly selected courses. As indicated, we also fit linear models to each of these data sets. Each course presented here exhibits a decline in participation over time, and the rest behaved similarly as well. Fig. 3b shows the distribution of the decline rate, or the slope of the models, for all the courses (mean $= -5.0$ and sd $= 8.7$; 72 of 73 are negative[4]). The variables $\{z_{i,t}\}_{t\geq 1}$ are qualitatively similar.

We next study the distribution on the count differences between two consecutive days in the same course. It is not uncommon to see outliers due to large fluctuations in discussion, especially near the start date or when homework/exams are due. After we remove the top and bottom 3 percent outliers from each course, the count differences follow a Gaussian distribution in most cases; see Fig. 1b for the Q-Q plots. The five courses shown here all pass Shapiro's normality test, with p-values ≥ 0.01. Overall, 51/73 courses passed (see Table 1).

Video length. The mean video length across the courses is 12.71 hours (sd $= 7.85$). We further analyzed the breakdown of video length by categories, and did not see discrepancies between them.

Length of the courses. All courses in our data set are between 4 and 14 weeks long. The mean length is 58.8 days (sd $= 15.3$).

Homework assignments. The mean number of staff-graded assignments per course is 10.13 (sd $= 10.88$). Out of the 73 courses, six of them did not have any staff-graded assignments. A total of 39 courses had peer-graded homework; among these, five were vocational courses, 11 were science or applied science, and 23 were humanities and social science.

Staff activity. On average, there were 366.9 posts in each course made by the teaching staff (sd $= 446.1$). Two courses had no staff posts.

Postulation of a model. Based on the evidence presented above, we postulate the following linear model for the post counts. Let $y_{i,t}$ be the number of posts on the tth day in the ith course. We assume $y_{i,t+1} - y_{i,t} \sim \mathcal{N}(\mu_i, \sigma_i)$, i.e., $y_{i,t} = \sum_{j\leq t} \mathcal{N}(\mu_i, \sigma_i) = \mathcal{N}(t\mu_i, \sqrt{t}\sigma_i)$. Here, the mean term grows linear in t while the "noise term" grows linear in $\sqrt{t}$. When t is sufficiently large, the mean term dominates $y_{i,t}$.

In other words, we may model $y_{i,t} = A_i t + B_i + \epsilon_{i,t}$, where A_i and B_i only depend on the factors of the ith course. Note that while serial dependency may be present, we believe this factor-adjusted, deterministic, linear trend is sufficient to explain the pattern; this is confirmed by our subsequent empirical results.

3.1.2 Regression Model

We will now present our linear model. Concretely, the number of posts $y_{i,t}$ is linearly related to the course factors, the variable t, and the interacting terms between t and the factors, as

$$y_{i,t} = At + B, \tag{1}$$

where $A = \beta_1 Q_i + \beta_2 V_i + \beta_3 L_i + \beta_4 D_i + \beta_5 P_i + \beta_6 S_i + \beta_7 H_i + \beta_8 M_i + \beta_{16}$ and $B = \beta_0 + \beta_9 Q_i + \beta_{10} V_i + \beta_{11} L_i + \beta_{12} D_i + \beta_{13} P_i + \beta_{14} S_i + \beta_{15} M_i$. Thus, we can view the parameters $\beta_1, \ldots, \beta_8$, and β_{16} as being related to the participation decline rate, and $\beta_0, \beta_9, \ldots, \beta_{15}$ as to the initial participation volume.

We fit these parameters to our data set using ordinary least-squares regression, and the second column in Table 2 shows the results (for $y_{i,t}$). We make a number of observations:

- It is evident that there is a significant relationship between the number of posts $y_{i,t}$ and the intrinsic

3. Choosing 2 as the threshold is rather arbitrary. The goal here is to show that many students make a small number of posts.

4. The only course with a positive rate also has a negative rate when we count the number of distinct users instead.

TABLE 1
List of Course Codes and Their Full Names

Code	Full name	Code	Full Name
aboriginaled-001★	Aboriginal World Views And Education	intropsych-001★	Introduction To Psychology
adhd-001★	"Pay Attention!!" Adhd Through The Lifespan	introstats-001★△	Statistics: Making Sense Of Data
analyticalchem-001★△	Analytical Chemistry	latinamericanculture-001⊥	Latin American Culture
ancientgreeks-001★△	The Ancient Greeks	lawandecon-001★	Property And Liability: An Introduction To Law And Economics
art-001★△	Introduction To Art: Concepts and Techniques	lead-ei-001★	Inspiring Leadership Through Emotional Intelligence
audiomusicengpart1-001★△	Fundamentals Of Audio And Music Engineering	lyingcamera-001★	The Camera Never Lies
behavioralecon-001★⊥	A Beginner'S Guide To Irrational Behavior	macroeconomics-001★△	Principles Of Macroeconomics
bioelectricity-002★△	Bioelectricity: A Quantitative Approach	malsoftware-001★△	Malicious Software And Its Underground Economy
bluebrain-001★	Synapses, Neurons, And Brains	medicalneuro-001★△	Medical Neuroscience
climateliteracy-001★△	Climate Literarcy: Navigating Climate Change	mentalhealth-002★△	Mental Health And Illness
compfinance-004	Introduction To Computational Finance And Financial Econometrics	ml-003★⊥	Machine Learning
compilers-2012-002★△	Compilers	modernpostmodern-001★	The Modern And The Postmodern
compilers-003	Compilers	molevol-001★△	Computational Molecular Evoluation
compstrategy-001	Competitive Strategy	mosfet-001★△	Mos Transistors
crypto-007★△	Cryptography 1	mythology-002★△	Greek And Roman Mythology
datasci-001★⊥	Introduction To Data Science	networksonline-001★△	Social And Economic Networks: Models And Analysis
digitalmedia-001★△	Creative Programming For Digital Medial And Mobile App	neuralnets-2012-001★△	Neural Networks For Machine Learning
drugdiscovery-001★△	Drug Discovery, Development and Commercialization	newwayhealthcare-001★	Interprofessional Healthcare Informatics
ecfoodandyou-001★△⊥	Economic Issues, Food, and You	nlangp-001★△	Natural Language Processing
einstein-001★△	Understanding Einstein: The Special Theory Of Relativity	nuclearscience-001★△	A Look At Nuclear Science And Technlogy
engcomlaw-001★△	English Common Law: Structure And Principles	nutrition-002★	Nutrition For Health Promotion An Disease Prevention
epigenetics-001	Epigenegic Control Of Gene Expression	oldglobe-001★△	Growing Old Around The Globe
fe-001★	Financial Engineering And Risk Management	operations-002★	An Introduction To Operations Management
friendsmoneybytes-002★△	Networks: Friends, Money, And Bytes	orgchem2a-001★△	Intermediate Organic Chemistry—Part I
gametheory-2012-002★	Game Theory	pgm-003★△	Probabilistic Graphical Models
genchem1-001★△	Chemistry: Conceptual Development And Application	posa-001★△⊥	Pattern-Oriented Software Architectures For Concurrent And Networked Software
globalenergy-001★	Global Sustainable Energy: Past, Present And Future	progfun-002★△⊥	Functional Programming Principles In Scala
globalfoodsystems-001★△	Sustainability Of Food Systems: A Global Life Cycle Perspective	programdesign-001★△	Introduction To Systematic Program Design
gtcomp-001★△	First-Year Composition 2.0	programming1-2012-001★⊥	Learning To Program: The Fundamentals
hci-003★△	Human-Computer Interaction	programming2-001★△	Learn To Program: Crafting Quality Code
healthforall-002★△⊥	Health For All Through Primary Health Care	rationing-001★△	Rationing And Allocating Scarce Medical Resources
healthinformatics-001★△	Health Informatics In The Cloud	rosc-001★△	Cardiac Arrest, Hypothermia, And Resuscitation Science
healthpolicy-002★△	Health Policy And The Affordable Care Act	sciwrite-2012-001★△⊥	Writing In The Sciences
historyofrock1-001★	History Of Rock, Part One	sdn-001★△	Software Defined Networking
hollywood-001★	The Language Of Hollywood: Storytelling, Sound, And Color	secrets-001★△	Archaelology's Dirty Little Secrets
hwswinterface-001★	The Hardware/Software Interface	socialepi-001★△	Social Epidemiology
images-2012-001★△	Image And Video Processing: From Mars To Hollywood With A Stop At The Hospital	sustainableag-001★△	Sustainable Agricultural Land Management
intlcriminallaw-001★△	Introduction To International Criminal Law	usefulgenetics-001★△	Useful Genetics
introeulaw-001★△	The Law Of The European Union: An Introduction	wealthofnations-001★△	Generating The Wealth Of Nations

All 78 were used in the forum analysis. A ★ indicates a course that was used in the statistical analysis (73/78), a △ indicates one that passed the Shapiro test (51/78, see Section 3), and a ⊥ indicates one that was used for testing in topic extraction and relevance ranking (10/78).

TABLE 2
Regression Parameters for Forum Activity Models, with Levels of Significance Indicated for Each

	On $y_{i,t}$	On $z_{i,t}$	On $\log(z_{i,t})$
(Intercept)	18.276	70.252^{***}	4.268^{***}
$Q_i t$	1.511^{***}	0.847^{***}	0.014^{***}
$V_i t$	3.328^{***}	1.463^{***}	0.011^{**}
$L_i t$	-0.071^{***}	-0.024^{***}	
$D_i t$	0.034^{***}	0.025^{***}	0.001^{***}
$P_i t$	-0.631^{**}	-0.375^{***}	0.003
$S_i t$	-0.168^{**}	-0.067^{**}	-0.001^{**}
$H_i t$	0.000	-0.001	
$M_i t$ (or $M_i' t$)	-0.007^{***}	-0.005^{***}	0.000^{**}
Q_i	-13.975	-23.737^{***}	-0.185^{**}
V_i	-135.567^{***}	-61.404^{***}	-0.153
L_i	1.960^{**}	-0.049	
D_i	-0.561	-0.624^{***}	-0.010^{***}
P_i	88.289^{***}	32.005^{***}	0.247^{***}
S_i	6.050^{**}	3.249^{***}	0.074^{***}
H_i	1.398^{**}	0.973^{***}	
M_i (or M_i')	0.481^{***}	0.360^{***}	0.003^{***}
t	-1.864^{***}	-1.980^{***}	-0.071^{***}
R^2	0.555	0.467	0.530
Adj. R^2	0.554	0.465	0.526
Num. obs.	5074	5074	1711

$^{***}p < 0.01$, $^{**}p < 0.05$, $^{*}p < 0.1$.

popularity M_i, but any impact of M_i on the decline rate in the long run appears very light.

- The coefficients $Q_i, V_i, Q_i t$, and $V_i t$ for quantitative and vocational courses suggest that while they are associated with a lower volume of forum participation initially, in the long run they tend to experience lower decline rates (all p-values $\leq 10^{-6}$).
- The results for the number of teaching staff posts S_i are surprising: while teaching staff active participation in the forum is associated with a higher volume of discussion (for every additional post by the teaching staff, there are on average 6.05 additional posts in the forum *each day*), in the long run it does not seem to reduce the decline rate. In fact, there is evidence that an increase in staff participation is correlated with a *higher* decline over time (p-value $= 0.021$).
- The relationship between $y_{i,t}$ and the number of peer-reviewed homeworks P_i is similar: while the presence of peer-reviewed homework is associated with 88.29 additional posts per day on average, it is also correlated with a higher decline rate (p-value $= 0.018$).

We remark that the p-value of the model is 2.2×10^{-16}, suggesting overall significance. Also, we diagnosed the residuals, which did not seem to elicit any heteroscedastic pattern. We further checked the differences between the slope of t (i.e., the quantity A_i in Equation (1)) under our proposed multivariate regression model and the counterpart for univariate regression with a single parameter imposed on t for each course (i.e., the slopes computed as in Figure 1a); these differences were reasonably small in magnitude.

Next, we move to the model for $z_{i,t}$, the number of distinct forum users in each course each day. We used the same format and set of regressors as in Equation (1), except that we substituted the intrinsic popularity M_i with M_i'. The results are shown in the second column of Table 2, and we can see that the relationships are qualitatively similar: quantitative and vocational courses (Q and V) are associated with a smaller volume of distinct users on the forums initially, but also with smaller decline rates over time (all with p-values $\leq 10^{-6}$); teaching staff participation (S) is associated with an increased number of distinct users on the forums (p-value $= 0.00994$), but also with higher decline rates (p-value $= 0.038$); and the presence of peer-grading (P) is associated with an increase in the total number of distinct users (p-value $= 5.94 \times 10^{-10}$), but also with higher decline rates (p-value $= 0.0016$). Finally, we remark that the p-value of this model is 2.2×10^{-16} which again suggests overall significance, and that the residuals do not show any obvious patterns here either.

More robust linear model. While these linear regression models are in general robust against noise, we also performed analysis on the subset of courses whose residuals exhibited normality (the ones with "nice" behavior) to see whether or not the conclusions were consistent. Specifically, we chose the 51 courses whose count difference in posts passed the Shapiro test with p-value ≥ 0.01 after removing the top and bottom 3 percent outliers (see Table 1). We used the same format and regressors as the previous model, except we fit the data to the logarithm of the number of distinct users $\log(z_{i,t})$; this transformation was performed because it resulted in higher model significance (note that this transformation is not strictly necessary because the results were otherwise similar).

The third column of Table 2 presents our results. Since the variables for video length and graded homework (L_i, $L_i t$, H_i, and $H_i t$) are not statistically significant, we removed them from the model. The conclusions we see here are mostly consistent with those made previously, the exception being the terms involving the staff participation S_i: while staff participation is still associated with an increased number of distinct users each day on average, the correlation with the decline rate is negligible (coefficient $= -1.47 \times 10^{-3}$, p-value $= 0.04$).

We performed further analysis to test the residuals for normality. The p-value of the Shapiro test is 0.148, which indicates that our model fits well for these 51 courses.

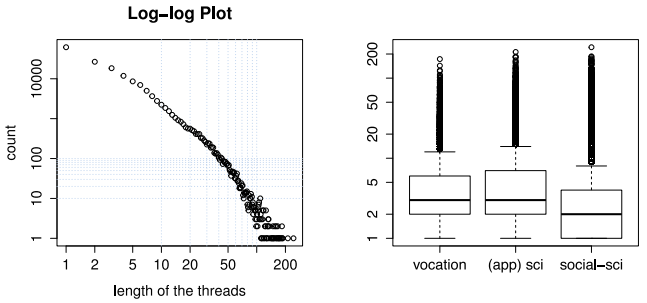
3.2 Attention Received by Each Thread

We now investigate the following question: Will the creation of more threads concurrently reduce the average "attention" that each receives on average?

Similar to [19], [28], we use the number of posts in a thread to measure the attention it receives. Note that there are other possible metrics for this, such as the total number of views or votes on each thread. But such information is not publicly available on Coursera.[5]

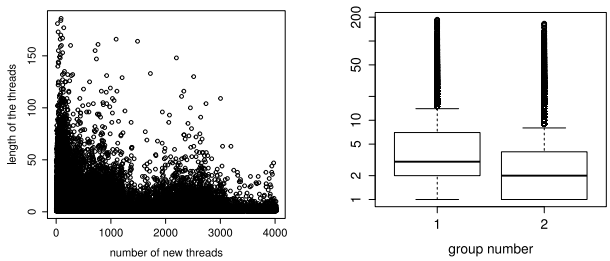
Thread length distribution. Before statistical testing, we briefly present the distribution of the thread lengths in the forums. Fig. 5a gives the distribution in log-log scale over

5. As of 2013, the number of views on each page is only available to course instructors. Also, the forums on Coursera present the *difference* between the number of up-votes and down-votes for each post; the total number of votes cannot be determined from this.



(a) Log-log plot of the thread length distribution. (b) Thread length by course category.

Fig. 5. Distribution of thread lengths over all 73 courses.



(a) Thread length vs. number of new threads. (b) Boxplots for the two groups (G_1: ≤ 140 new threads; G_2: otherwise)

Fig. 6. Discussion thread length against the number of threads created around the same time, using a window $t = 1$.

all 73 courses; the mean is 4.98 (median $= 2$ and sd $= 8.65$). Fig. 5b gives boxplots of the thread lengths by course category.

Independent two-sample t-test. We use a t-test to understand whether having more newly created threads will dilute the attention that each receives. Let h_j refer to a discussion thread within a given course, and let ℓ_j be the length of h_j as defined previously. Also, let $f(h_j, t)$ be the total number of other threads in this course that were created within the window of t days before and after h_j was created; for example, if h_j was created on July 2 at 3 pm, then $f(h_j, 1)$ is the number of threads besides h_j that were created between July 1 at 3 pm and July 3 at 3 pm.

Fig. 6a shows the plot of ℓ_j against $f(h_j, 1)$ for all threads in our data set. We first attempt to fit this with a linear model, but find its explanatory power quite low (R^2 is below 0.02). As a result, we resort to two-sample procedures and partition the threads into two groups. The first, G_1, contains all the threads h_i such that $f(h_i, 1) \leq 140$, and the second, G_2, contains the rest. The threshold number 140 is chosen so that the size and variances were within a factor of two between the groups (size of G_1 is 44,971, $\mathrm{Var}_{h_i \in G_1}(\ell_i) = 103.94$; size of G_2 is 76,890, $\mathrm{Var}_{h_i \in G_2}(\ell_i) = 62.14$). We shall refer to G_1 as the small group and G_2 as the large group; Fig. 6b gives the boxplot of the log of thread lengths in both groups.

Our null and alternative hypotheses are as follows:

- H_0. The small group's thread length is no greater than the large group's thread length.
- H_1. The small group's thread length is greater than the large group's thread length.

The comparison above is understood to be with respect to some central tendency measure. A t-test yielded a t-statistic of 40.3 and a p-value $\leq 2.2 \times 10^{-16}$. We also carried out a Mann-Whitney U-test, which yielded a similar p-value $\leq 2.2 \times 10^{-16}$. Both tests indicate that we can reject the null hypothesis with high confidence, with respect to the mean and median, respectively. Therefore, there is strong evidence that the creation of more threads concurrently is correlated with a reduction in the attention that each thread receives.

4 A Generative Model

Now that we have explored the first research question, we move on to Q2. The goal here is to present a generative model for thread discussions that can help guide user navigation on MOOC forums. Recall from the discussion and data analysis in Sections 1 and 2 that these forums typically contain a substantial amount of small-talk and less course-relevant threads, especially in the initial stages of a course.

For motivation, we will first present an evaluation of standard classification algorithms for filtering small-talk. Specifically, we will consider naive Bayes (NB) and support vector machine (SVM) classifiers, each of which are known to be effective in this problem space. While the SVM classifier shows reasonable performance in our analysis, the NB classifier experiences excessively high false positive rates. We leverage the clues from this discrepancy in performance to design a generative model; our model explains this discrepancy and guides us in the design of topic extraction and relevance-ranking algorithms (Section 5).

4.1 Understanding the Classifiers

Thread labeling. As discussed in Section 2, we used MTurk to label threads as small-talk or not. Our sample consisted of (i) a random selection of 30 threads from each course, and (ii) the first 30 even-numbered threads from a random selection of 40 courses. (iii) Allowed us to obtain more samples from the small-talk category.

We next split the sample into a training set and a test set: each thread became a training point with probability 0.85, and a test point otherwise. Since the test set was smaller, the authors made an extra pass through it to further reduce any errors in the labels. k-fold cross validation was not performed because the test set appeared to be sufficiently large.

Classifier training. The NB algorithm was implemented based on [29], [30]. We took two approaches: (1) training a single classifier using training data over all courses and applying it to the entire test set; and (2) training one classifier for each course and applying them to the test data from their respective courses. For SVM, we used the open source SVM Light software [31] with a linear Kernel and its default parameters.

Results. We compare performance of the algorithms on the test data. To do so, we measured the true and false positive rates, which are more appropriate to measure than precision and recall in this scenario (see [24] for a discussion). Here, the true positive rate (or sensitivity) is the fraction of small-talk threads that were correctly classified, and the false positive rate (or fall-out) is the fraction of course-related threads that were incorrectly classified as small-talk.

	true positive rate	false positve rate
NB (aggregate)	96.7%	35.9%
NB (separate)	96.7%	76.8%
SVM ($\tau = -1.015$)	93.3%	17.3%
SVM ($\tau = -0.995$)	86.7%	5.9%

Fig. 7. Comparing the SVM and the NB classifier.

Fig. 7 shows a comparison between SVM and NB. Both the true and false positive rates were very high for NB, and separating the data by course almost doubled the fall-out for the same sensitivity. For SVM, we varied the threshold decision parameter τ to trade-off these two rates, where higher τ gives a higher true positive rate. Referring to the first row in Fig. 7, we see that SVM obtained a substantially better false positive rate than NB even when the true positive rates were similar. And if we allow for a smaller sensitivity, the false positive rate reduced even further, as shown in the last row.

We remark that an advantage of our classification process is that we do not need a large number of features to obtain sufficient results.

4.2 A Unified Topical Model

Distributions. Let C be the set of n words that appear across all threads. Our model consists of the following distributions on C:

- *Background distribution* ($\mathcal{B}$). This models (in terms of probability density) the set of commonly used words in the English language (but not so common as to appear in a list of stop words), and is not topic- or course-dependent.
- *Small-talk and logistics topical distribution* ($\mathcal{T}_0$). This models keywords that are more likely to appear in small-talk or logistics discussions, and is not course-dependent.
- *Course-specific topical distribution* ($\mathcal{T}_i$, *for each course* i). This models keywords that are more likely to appear in course-specific discussions in the ith course.

Sampling. A thread in the ith course is sampled such that with probability p_i, the thread is logistic/smalltalk; otherwise, the thread is course-specific. Here, p_i can be different for different courses.

When a thread is logistic/smalltalk, we model its constituent words as being independent and identically distributed (i.i.d.) samples from $\mathcal{D}_0(i)$, which defined is a convex combination of $\mathcal{B}$ and $\mathcal{T}_0$: with probability $(1-\epsilon)$, a word is sampled from $\mathcal{B}$; otherwise, it is sampled from $\mathcal{T}_0$. Notice $\mathcal{D}_0(i)$ is the same across all courses i. On the other hand, when a thread is course-specific, the words are modeled as i.i.d. samples from $\mathcal{D}_1(i)$, which is defined as: with probability $(1-\epsilon)$, a word is sampled from $\mathcal{B}$; otherwise, it is sampled from $\mathcal{T}_i$. Further, for exposition purposes we make the following assumptions (most of which can be relaxed):

Near-uniformity in $\mathcal{B}$. For any $w \in C$, we assume $\Pr_{\mathcal{D}_j(i)}(w) = \Theta(\frac{1}{n})$ for each i and j. Here, we shall imagine that C represents the words that are outside of a stopword list but cover important topics in different courses. This assumption is justified by the heavy tail distribution over words in the English language.

Distinguishability of the topics. Let $\mathrm{Supp}(\mathcal{D})$ be the support of a distribution $\mathcal{D}$. For each i and j, we assume $\mathrm{Supp}(\mathcal{T}_i)$ and $\mathrm{Supp}(\mathcal{T}_j)$ do not overlap (the supports of $\mathcal{D}_1(i)$ and $\mathcal{D}_1(j)$ may still), meaning the keywords are mutually exclusive. Furthermore, for any $w \in \mathrm{Supp}(\mathcal{T}_i)$, we assume $\ell \leq \frac{\Pr_{\mathcal{D}_1(i)}(w)}{\Pr_{\mathcal{B}}(w)} \leq u$, where $1 < \ell < u$ are two constants, implying that any keyword specific to course i is more probable within $\mathcal{D}_1(i)$ than $\mathcal{B}$.

Classifier behavior. The following theorem explains the behavior of the NB and SVM classifiers:

Theorem 4.1. *For the generative model presented above, there exists an ϵ, $p_1, \ldots, p_m$, and a sequence $b_1, \ldots, b_m$, such that if b_i training samples are obtained for the ith course, then:*

1) *With constant probability over the training set, NB will have poor performance for some courses (i.e., with high probability (whp), the classifier errs on the negative threads) regardless of whether it is trained per course or across all courses.*
2) *There exists a good separation plane for SVM so that whp a discussion thread will be classified correctly.*

Before giving the proof of this theorem, we will present a high level summary of the first part. When we train the NB classifier per course, there is insufficient training data (e.g., $\approx$30 threads for each course), which causes the conditional probabilities on D_0 to be overestimated for each word with constant probability. On the other hand, when we train the classifier over all courses, it cannot address the fact that p_i is different for each course and will thus use inaccurate prior information in developing the classifier for some of them. In these courses, the classifier may have poor performance.

Proof of Theorem 4.1. For exposition purpose, let us assume that each thread has uniform size and contains s words. We shall first show part 2, i.e., there exists a separating plane for smalltalk and non-smalltalk type threads. Let $S_0 = \mathrm{Supp}(\mathcal{T}_0)$ be the support of $\mathcal{T}_0$. Let $\vec{a} = (a_1, \ldots, a_n)$ be a binary vector such that $a_i = 1$ if and only if $i \in S_0$.

Now for any i, consider the distributions $\mathcal{D}_0(i)$ and $\mathcal{D}_1(i)$. We can see that

$$\mathrm{E}_{\mathcal{D}_0}[w \in S_0] = \epsilon + c_0\epsilon \quad \text{and} \quad \mathrm{E}_{\mathcal{D}_1}[w \in S_0] = c_0\epsilon$$

for some constant c_0. Thus, for any document W of size s, we have

$$\mathrm{E}_{\mathcal{D}_0^s}\left[\sum_{w\in W}(w \in S_0)\right] = s(1+c_0)\epsilon,\ \mathrm{E}_{\mathcal{D}_1^s}\left[\sum_{w\in W}(w \in S)\right] = c_0 s\epsilon.$$

We may set the threshold as $\tau = s(\frac{1}{2} + c_0)\epsilon$, i.e., for any thread W (in the bag-of-word representation), we classify it as a logistic/smalltalk thread if and only if $aW\tau$. By using a simple Chernoff bound, we see that

$$\Pr_{\mathcal{D}_0^s}[aW \geq \tau] = \Pr_{\mathcal{D}_0^s}\left[\sum_{w\in W}(w \in S_0) \geq \tau\right] \geq 1 - \frac{1}{n},$$

and $\Pr_{\mathcal{D}_0^s}[aW \leq \tau] \geq 1 - \frac{1}{n}$ when $s\epsilon = \Omega(\log n)$. Thus, this separation plane works well in SVM.

Next, let us move to prove the first part of the theorem.

We now formalize this intuition. Let us start with the case where we train the classifier per course. Let us focus on the first course. Here, we set $s_1 = \sqrt{n}$, $p_1 = c \log n/\sqrt{n}$ for a suitably large constant, and $b_1 = \frac{1}{p_1}$. Furthermore, we adopt the standard approach to initialize the estimates on the conditional probability, i.e., $\hat{\Pr}[w \mid \mathcal{D}_0] = \hat{\Pr}[w \mid \mathcal{D}_1] = 1$ for all words w.

We can see that from b_1 samples, with constant probability that we see only $O(1)$ positive threads (i.e., small-talk type threads) sampled from $\mathcal{D}_0$. In this case, the conditional estimates in the Naive Bayes classifiers are

$$\hat{\Pr}[w \mid \mathcal{D}_0] = \Theta(1) \quad \text{and} \quad \hat{\Pr}[w \mid \mathcal{D}_1] = \Theta(\frac{1}{n}).$$

Also, we have $\hat{\Pr}[\mathcal{D}_0] = O(p_1)$. Let W be an arbitrary negative thread. Recall that $\hat{\Pr}[W \in \mathcal{D}_i \mid W] = \prod_{w \in W} \hat{\Pr}[w \mid \mathcal{D}_i]\hat{\Pr}[\mathcal{D}_i]$, W is classified as positive if and only if

$$\sum_{w \in W} \log\left(\hat{\Pr}[w \mid \mathcal{D}_0]\right) + \log \hat{\Pr}[\mathcal{D}_0] > \sum_{w \in W} \log \hat{\Pr}[w \mid \mathcal{D}_1] + \log \hat{\Pr}[\mathcal{D}_1].$$

We can see that (again from Chernoff bounds) with high probability the left-hand side $= \Theta(-s - \log n)$ while the right-hand side $= \Theta(-s \log n)$. Thus, with constant probability over the training data the Naive classifier will mis-classify the negative threads with high probability (over the testing data).

We now move to the case we train the classifier from threads across all the courses. We need only two courses to construct the negative example. Here, we use the same distributions and parameter for the first course as we did in the per-course-classifier analysis. We now specify the second course. We let $s_2 = n^d$ for a constant d to be decided later and let $p_2 = (1 - \frac{1}{n^d})$. Under this setting, with high probability we see only a constant number of threads from $\mathcal{D}_1(2)$. Thus $\hat{\Pr}[\mathcal{D}_0] = O(1/n^d)$, $\hat{\Pr}[w \mid \mathcal{D}_0]$ are all accurate (within a small multiplicative error) while $\hat{\Pr}[w \mid \mathcal{D}_1]$ is also statistically close to $\mathcal{D}_1(1)$ (because we only see a constant number of samples from $\mathcal{D}_1(2)$). Thus, when W is a negative sample, we have

$$\sum_{w \in W} \log \hat{\Pr}[w \mid \mathcal{D}_1] + \log \hat{\Pr}[\mathcal{D}_1] - \sum_{w \in W} \log \hat{\Pr}[w \mid \mathcal{D}_0] + \log \hat{\Pr}[\mathcal{D}_0]$$

$$\approx \Theta(\epsilon \ell) - d \log \frac{1}{n} < 0,$$

for a sufficiently large d. Here "$\approx$" refers to the sum of r.v.s concentrates around the right hand side. In other words, with constant probability the classifier fails to work for most of the negative samples. □

5 Topic Extraction and Ranking

We will now apply the generative model presented in the previous section to extract keywords from forum discussions and to design a relevance-ranking algorithm.

5.1 Topic Extraction Algorithm

We start with a simple algorithm for extracting $\mathrm{Supp}(\mathcal{T}_i)$, the set of course-specific words for the ith course. *Motivation for topic extraction.* Knowledge of $\mathrm{Supp}(\mathcal{T}_i)$ is a convenient way for students in course i to identify key topics. One can imagine some basic approaches for determining the keywords, such as using the those found on a course syllabus or simply asking the instructor to list them directly. We argue that these methods are not suitable solutions here, for a number of reasons.

First, while the keywords in the course syllabus may appropriately summarize information contained in the instructors' lectures, they will not always do so for the course *discussions*, with the specific points that students have focused on. These discussions may even deviate from the focus of the lectures. This point will be illustrated through examples below. Second, when the same course is offered multiple times, the lecture material will likely remain static from one instance to the next. On the other hand, one would expect the topics of the forum discussion to vary (e.g., with a different set of students or to reflect current events), causing $\mathcal{T}_i$ to change. Finally, as we will see, a topic-extraction algorithm can help rank the relevance of each thread; the number of topical keywords each one contains gives an understanding as to how relevant each one is.

The topic extraction algorithm. Our algorithm uses the following two parts as training data for course i:

- *Background training data*. This consists of all the forum discussions for k courses.
- *Course-dependent training data*. This consists of forum discussions in the first few days (approximately 10) of the ith course.

Now, let n be the total number of words in the background training set, and let $\hat{\mathcal{D}}_n$ be the empirical unigram distribution associated with both the background and course-dependent training data. Let $\hat{\mathcal{E}}$ be the empirical distribution associated with the course-dependent training data, and let $W = \{w_1, \ldots, w_\ell\}$ be the support of $\hat{\mathcal{E}}$. Let w denote an arbitrary word in W, $p_{\hat{\mathcal{D}}}(w)$ the probability mass of w under the distribution $\hat{\mathcal{D}}_n$, and $p_{\hat{\mathcal{E}}}(w)$ the probability mass of w under $\hat{\mathcal{E}}$. We define the "surprise weight" of w as

$$\gamma(w) = \frac{p_{\hat{\mathcal{E}}}(w)n}{\sqrt{p_{\hat{\mathcal{D}}}(w)n}} = \frac{p_{\hat{\mathcal{E}}}(w)\sqrt{n}}{\sqrt{p_{\hat{\mathcal{D}}}(w)}}. \tag{2}$$

Intuitively, this will assign higher weights to keywords whose empirical probability mass substantially deviates from the baseline distribution.

The words in W are ranked according to the $\gamma(\cdot)$ function so that the first word in the resulting list is the one with the highest value. Our keyword summary is then the top-k (ordered list of) words in the ranking. We have the following Corollary:

Corollary 5.1. *Under the generative model presented and assuming $p_i = \Theta(1)$ and $k = |\mathrm{Supp}(\mathcal{T}_i)|$ is known, the topical extraction algorithm will successfully identify $\mathrm{Supp}(\mathcal{T}_i)$ when the training data is sufficiently large.*

Experiments. We now evaluate the efficacy of our topical algorithm. Here, the main discovery is that we only need

Course name	Keywords
Machine Learning	theta octave regression gradient descent matrix ex1 alpha machine vector function linear computecost gnuplot data
Health for All Through Primary Health Care	health phc care primary healthcare community ata alma perry henry medical clinics services public communities
Pattern-Oriented Software Architectures	doug vanderbilt patterns dre schmidt posa concurrent concurrency gof middleware corba frameworks pattern software singleton
Latin American Culture	latin america culture indigenous american cultures democracy spanish imposed political cultural traditions countries democratic mexico

Fig. 8. Top 15 keywords extracted by the topical algorithm for four of the courses.

approximately 10 days of course-dependent training data for the algorithm to accurately identify $\mathcal{T}_i$.

We tested the algorithm on 10 random courses (see Table 1) out of those that had larger discussion volume. To select this set, for each of the 78 courses we first counted the number of days in which 15 or more new threads were created; call this number d_i for the ith course. We then found all courses with $d_i \geq 25$. There were 24 such courses, and we randomly chose 10 of these for testing. We then chose 50 among the remaining 68 courses for background training.

The topical algorithm was quite effective in identifying the keywords across all 10 test courses; Fig. 8 shows the results for four of the courses, focusing on the top-15 keywords for each. We remark that the terms with the highest inverse document frequency (idf) scores were mostly meaningless. As a comparison, we looked at which of these keywords were contained in the respective course syllabi, and noted that many of the terms corresponding to specific discussion information were not present. For example, in Machine Learning, "computecost" and "gnuplot" are application-specific terms that the students spent substantial time discussing in relation to programming assignments. These terms do not appear in the syllabus for the course, nor would one expect them to since they constitute low-level details. Similarly, for Pattern-Oriented Software Architectures, common object request broker architecture (corba) is a programming standard that students focused on in the forums, but is too specific for the syllabus.

We also examined the convergence rate, which corresponds to the window of time needed for our algorithm to "warm up". The sets of top-50 keywords quickly stabilized for the 10 courses; Fig. 9 presents the normalized Kendall tau distance (defined as the Kendall tau distance divided by the total number of distinct word pairs) for keyword ranks between two consecutive days, using the same set of courses as in Fig. 8. The distances converge to below 2 percent after approximately 10 days, which is a reasonable waiting period.

5.2 Ranking Algorithm

We now leverage the topic-extraction algorithm to design a relevance-ranking algorithm for discussion threads. To begin, we will walk through a hypothetical scenario to set up the notation.

Suppose Alice enrolled in a machine learning course and used the forum on the 12th day. Then on the 15th day, she logs in again. Our goal here is to rank the threads that were created between the 12th and the 15th day so that Alice can be directed to the most relevant discussions. In this example, we refer to the first 12 days as the *warming period*, and the 12th-15th as the *query period* for the ranking algorithm. Finally, all 15 days constitute the *window of interest*.

The relevance-ranking algorithm. Roughly speaking, our ranking algorithm assigns more weight to threads that contain more keywords from $\mathrm{Supp}(\mathcal{T}_i)$. More specifically, it consists of two steps:

Step 1: Assigning weights. Let w be an arbitrary keyword that appears in a thread, and let $r(w)$ be the rank of w in the output of our topical algorithm for the top-50 keywords; in case w is not in this list, $r(w) = \infty$. The weight of w, $\eta(w)$, is given by $\alpha^{r(w)}$, where $\alpha \in (0, 1)$. We set $\alpha = 0.96$ in our experiments.

Step 2: Assigning scores. The score of a thread is simply the sum of the weight of its constituent words (with repetition); formally, if F_j is the list of words in thread j, then the score of j is given by

$$\mathcal{S}(j) = \sum_{w \in F_j} \eta(w) = \sum_{w \in F_j} \alpha^{r(w)}. \tag{3}$$

The threads are ranked in descending order according to the $\mathcal{S}(\cdot)$ function.

While our ranking algorithm is technically simple, we emphasize the primary goal here is a proof-of-concept for the efficacy of our generative model.

Baseline algorithms. We compare our algorithm with two natural baselines, one term-based and one "random walk"-based.

tf-idf algorithm. We use tf-idf [32] as the term-based algorithm, treating each thread as a document. The score of each

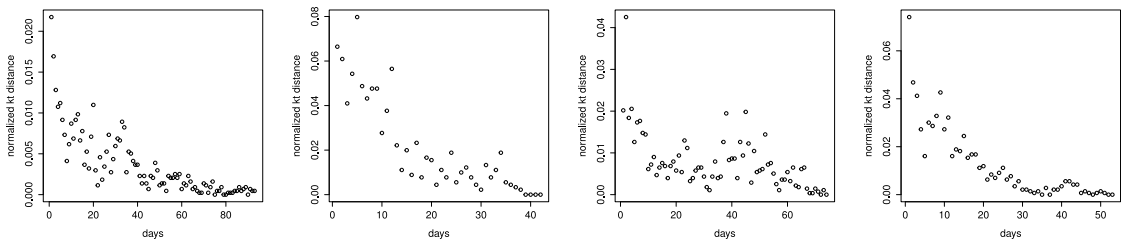


Fig. 9. Topical algorithm's convergence rate for the four courses in Fig. 8, with "Machine Learning" to "Latin American Culture" from left to right.

thread is computed by summing the tf-idf values of its constituent words; the threads are subsequently ranked in descending order based on these scores.

Kleinberg HITS-type algorithm. For the random walk-based algorithm, we use a variation of HITS, which is effective at finding the "importance" of nodes in a network. Our implementation of HITS works as follows. We construct a bipartite graph where each node on the left represents a user and each on the right represents a thread. A user is connected with a thread if and only if the user has posted or commented in the thread. We then interpret users as "hubs" and threads as "authorities", and apply the standard HITS updates [33]. Once the algorithm has terminated (when the ℓ_2 distances for authority and hub weights between two iterations are sufficiently small), threads are ranked in descending order based on their authority scores.

Difference from baselines. We highlight the differences between our topical algorithm and the two baselines:

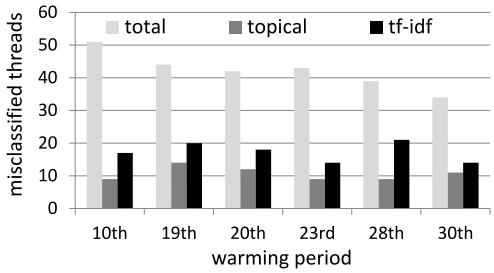
Comparing with tf-idf. At a high level, our algorithm is similar to tf-idf, because they both assign weights to each word that appears in a thread and sum these weights to reach a total score. However, there are two key differences. First, our algorithm considers all the threads in a course in extracting keywords while the tf-idf technique focuses on a single thread at a time. Thus, tf-idf can pick up threads that contain a series of low-frequency words which are irrelevant to the course but have high idf values. Two specific examples of this could be (1) a thread that includes discussion in non-English languages, and (2) a thread of people soliciting professional advice or help (e.g., a doctor's opinion on a person's sickness in a medical course). Second, along similar lines, the tf-idf technique is incapable of finding important keywords, because the terms with the highest idf scores are often those that appear exactly once. Most of these words are not very meaningful.

Comparing with HITS. HITS is a graph-based algorithm that can extract the popular threads from the forums. But the goal here is to find the relevant threads. Since HITS does not take into account the content in the discussions, we do not expect it to work as well in this application.

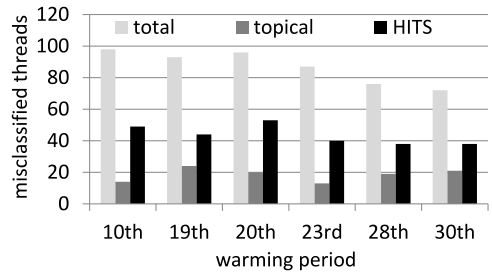
Evaluation. We will now experimentally test the intuition presented above. Specifically, our goal is to validate that our topical algorithm has higher performance in ranking threads than the tf-idf and HITS, because (1) tf-idf has a higher probability of misweighing non-relevant threads, such as non-English discussions, and (2) HITS ranks popularity instead of relevance.

We tested on the same 10 courses (see Table 1) with large discussion volume as we did to demonstrate keyword extraction. We used 10, and five randomly chosen integers between 10 and 30, as warming periods (in days), making six cases total for each course. For each of these warming periods, the query period was set to two days. Note that directly assessing the quality of multiple ranks efficiently in computational social choice is an open problem (see [34] and references therein); as a result, we focus on understanding the *differences* between our algorithm and the baselines, by comparing the number of *irrelevant* threads recommended by each.

To do so, for each course and each window of interest we extracted the the first 15 threads recommended by our algorithm, combined into a set S, and the first 15 recommended by each baseline, combined into a set S_b. Then, we found the set differences $D_1 = S - S_b$ (i.e., the threads recommended by our algorithm but not the baseline) and $D_2 = S_b - S$ (vice versa). Finally, we used MTurk to label whether the threads in D_1 and D_2 were relevant or not, which gave the difference in relevance counts between the algorithms.



(a) Comparing with tf-idf



(b) Comparing with HITS

Fig. 10. Relevance-ranking evaluation, by waiting period.

Result 1: Comparing with tf-idf. Here, $|D_1| = |D_2| = 253$ over the 10 courses and six days examined. sixty four were labeled as irrelevant in D_1, as opposed to 104 in D_2. Fig. 10a shows the breakdown of the misclassified threads by warming period (taking the total over all 10 courses). The light bar is the total size of the difference each day, the medium is the number of irrelevant threads from our algorithm, and the dark is the number of irrelevant threads from tf-idf. While the improvement is not dramatic, we can see that the topical algorithm is consistently better, validating our intuition. *Result 2. Comparing with HITS.* In this case, $|D_1| = |D_2| = 522$. One hundred and eleven threads were irrelevant in D_1, as opposed to 262 in D_2. Fig. 10b shows the breakdown by warming period; as one can see, the topical algorithm is again consistently better, and further, the difference is more substantial than in Result 1. This validates that the HITS algorithm is less effective in finding relevant threads.

6 Conclusion

We have investigated two of the issues present in MOOC discussion forums: sharp decline of participation over time, and information overload associated with the large number of threads. Through our analysis, we presented a large-scale statistical analysis of a MOOC platform (Coursera), in which we made a number of interesting observations; for instance, that active participation of the teaching staff is associated with an increase in discussion volume but does not reduce the participation decline rate. We also presented

two proof-of-concept algorithms for keyword extraction and relevance-ranking of discussion threads, each of which was demonstrated to be effective, through human evaluation when necessary.

The larger goal behind the two main research questions in this work is to improve the quality of learning via online discussion forums, by devising methods to sustain forum activities and to facilitate personalized learning. This paper makes a step in this direction by relying on an extensive empirical data set that allowed us to understand current user behavior and factors that could potentially change such behavior, as well as how to determine the most course-relevant discussions. Motivated by the results presented here, the open problems to be addressed next are how to reduce the decline of forum participation and how to leverage thread rankings to make effective individualized recommendations.

Acknowledgments

This work is in part supported by an ARO grant W911 NF-14-1-0190, a Princeton University grant, and a Guggenheim Fellowship. Additionally, the authors thank the reviewers for their valuable feedback.

References

[1] K. Swan, "Building learning communities in online courses: The importance of interaction," *Educ. Commun. Inf.*, vol. 2, no. 1, pp. 23–49, 2002.

[2] G. D. Kuh, "What student affairs professionals need to know about student engagement," *J. College Student Develop.*, vol. 50, no. 6, pp. 683–706, 2009.

[3] P. Bouchard, "Some factors to consider when designing semi-autonomous learning environments," *Electron. J. e-Learn.*, vol. 7, no. 2, pp. 93–100, 2009.

[4] T. Valjataga, H. Poldoja, and M. Laanpere, "Open online courses: Responding to design challenges," in *Proc. 4th Int. Netw.-Based Educ.*, 2011, pp. 68–75.

[5] R. Kop, "The challenges to connectivist learning on open online networks: Learning experiences during a massive open online course,"*Int. Rev. Res. Open Distance Learn.*, vol. 12, no. 3, pp. 19–38, 2011.

[6] M. Qiu, J. Hewitt, and C. Brett, "Online class size, note reading, note writing and collaborative discourse," *Int. J. Comput.-Supported Collaborative Learn.*, vol. 7, no. 3, pp. 423–442, 2012.

[7] M. J. Eppler and J. Mengis, "The concept of information overload: A review of literature from organization science, accounting, marketing, mis, and related disciplines," *Inf. Soc.: An Int. J.*, vol. 20, no. 5, pp. 325–344, 2004.

[8] B. De Wever, T. Schellens, M. Valcke, and H. Van Keer, "Content analysis schemes to analyze transcripts of online asynchronous discussion groups: A review," *Comput. Educ.*, vol. 46, no. 1, pp. 6–28, 2006.

[9] I. de Waard, A. Koutropoulos, N. Ozdamar Keskin, S. C. Abajian, R. Hogue, O. C. Rodriquez, and M. S. Gallagher, "Exploring the MOOC format as a pedagogical approach for mLearning," in *Proc. 10th World Conf. Mobile Contextual Learn.*, Beijing China, 2011, pp. 1–11.

[10] R. Kosala and H. Blockeel, "Web mining research: A survey," *ACM SIGKDD Explorations Newslett.*, vol. 2, no. 1, pp. 1–15, 2000.

[11] C. Piech, J. Huang, Z. Chen, C. Do, A. Ng, and D. Koller. (2012, Jul.). Tuned models of peer assessment in MOOCs [Online]. Available: http://arxiv.org/abs/1307.2579

[12] J. Ahn, C. Weng, and B. S. Butler, "The dynamics of open, peer-to-peer learning: What factors influence participation in the P2P university?" in *Proc. 46th Hawaii Int. Conf. Syst. Sci.*, 2013, pp. 3098–3107.

[13] R. F. Kizilcec, C. Piech, and E. Schneider, "Deconstructing disengagement: Analyzing learner subpopulations in massive open online courses," in *Proc. 3rd Int. Conf. Learn. Analytics Knowl.*, 2013, pp. 170–179.

[14] J. Cheng, C. Kulkarni, and S. Klemmer, "Tools for predicting drop-off in large online classes," in *Proc. Conf. Comput. Supported Cooperative Work Companion*, 2013, pp. 121–124.

[15] A.-F. Lui, S. C. Li, and S. O. Choy, "An evaluation of automatic text categorization in online discussion analysis," in *Proc. 7th IEEE Int. Conf. Adv. Learn. Technol.*, 2007, pp. 205–209.

[16] A. Ghosh and J. Kleinberg, "Incentivizing participation in online forums for education," in *Proc. 14th ACM Conf. Electron. Commerce*, 2013, pp. 525–542.

[17] E. Mendes Rodrigues and N. Milic-Frayling, "Socializing or knowledge sharing?: Characterizing social intent in community question answering," in *Proc. 18th ACM Conf. Inf. Knowl. Manage.*, 2009, pp. 1127–1136.

[18] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi, "Measuring user influence in twitter: The million follower fallacy," in *Proc. 4th Int. Assoc. Adv. Artif. Intell. Conf. Weblogs Social Media*, 2010, pp. 10–17.

[19] T. Sun, W. Chen, Z. Liu, Y. Wang, X. Sun, M. Zhang, and C.-Y. Lin, "Participation maximization based on social influence in online discussion forums," in *Proc. Int. Assoc. Adv. Artif. Intell. Conf. Weblogs Social Media*, 2011, pp. 361–368.

[20] S. Jain, Y. Chen, and D. C. Parkes, "Designing incentives for online question and answer forums," in *Proc. 10th ACM Conf. Electron. Commerce*, 2009, pp. 129–138.

[21] J. Zhang, M. S. Ackerman, and L. Adamic, "Expertise networks in online communities: Structure and algorithms," in *Proc. 16th Int. Conf. World Wide Web*, 2007, pp. 221–230.

[22] P. Jurczyk and E. Agichtein, "Discovering authorities in question answer communities by using link analysis," in *Proc. 16th ACM Conf. Inf. Knowl. Manage.*, 2007, pp. 919–922.

[23] L. A. Adamic, J. Zhang, E. Bakshy, and M. S. Ackerman, "Knowledge sharing and yahoo answers: Everyone knows something," in *Proc. 17th Int. Conf. World Wide Web*, 2008, pp. 665–674.

[24] F. M. Harper, D. Moy, and J. A. Konstan, "Facts or friends?: Distinguishing informational and conversational questions in social Q&A sites," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2009, pp. 759–768.

[25] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec, "Discovering value from community activity on focused question answering sites: A case study of stack overflow," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2012, pp. 850–858.

[26] K. Chai, C. Wu, V. Potdar, and P. Hayati, "Automatically measuring the quality of user generated content in forums," in *Porc. 24th Int. Conf. Adv. Artif. Intell.*, 2011, pp. 51–60.

[27] C. Callison-Burch, and M. Dredze "Creating speech and language data with amazon's mechanical turk," in *Proc. NAACL HLT Workshop Creating Speech Lang. Data Amazon's Mech. Turk*, 2010, pp. 1–12.

[28] V. Belk, S. Lam, and C. Hayes, "Cross-community influence in discussion fora," in *Proc. Int. Assoc. Adv. Artif. Intell. Conf. Weblogs Social Media*, 2012, pp. 34–41.

[29] T. M. Mitchell, *Machine Learning*, 1st ed. New York, NY, USA: McGraw-Hill, 1997.

[30] P. Harrington, *Machine Learning in Action*. Greenwich, CT, USA: Manning, 2012.

[31] T. Joachims, "Making large-scale support vector machine learning practical," in *Advances in Kernel Methods*, B. Schölkopf, C. J. C. Burges and A. J. Smola, Eds. Cambridge, MA, USA: MIT Press, 1999, pp. 169–184.

[32] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge, U.K.: Cambridge Univ. Press, 2008.

[33] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *J. ACM*, vol. 46, no. 5, pp. 604–632, Sep. 1999.

[34] H. Azari Soufiani, D. C. Parkes, and L. Xia, "Preference elicitation for general random utility models," in *Proc. 29th Conf. Uncertainity Artif. Intell.*, 2013, pp. 596–605.

Christopher G. Brinton (M '08) received the BSEE degree from the College of New Jersey (valedictorian and summa cum laude) and the master's degree in electrical engineering from Princeton University in May 2011 and 2013, respectively. He is currently working toward the PhD degree in electrical engineering at Princeton. His primary research interests inlcude data analytics and algorithms for adaptive educational systems and social learning networks. He is a MOOC instructor and coauthor of a textbook on social and technological networking which became an Amazon bestseller in Technology. He is a student member of the IEEE.

Mung Chiang (M'03, SM'08, F'12) is the Arthur LeGrand Doty professor of electrical engineering at Princeton University. His research on communication networks received the 2013 Alan T. Waterman Award from the US National Science Foundation, the 2012 Kiyo Tomiyasu Award from IEEE, and various young investigator awards and paper prizes. A Technology Review TR35 Young Innovator Award recipient, he created the Princeton EDGE Lab in 2009 to bridge the theory-practice divide in networking by spanning from proofs to prototypes, resulting in several technology transfers to industry and startup companies. He is the chairman of the Princeton Entrepreneurship Advisory Committee and the director of the Keller Center for Innovations in Engineering Education. His MOOC in social and technological networks reached about 200,000 students since 2012 and lead to two undergraduate textbooks and he received the 2013 Frederick E. Terman Award from the American Society of Engineering Education. He was named a Guggenheim fellow in 2014. He is a fellow of the IEEE.

Shaili Jain received the BS degree in mathematics and the BSE degree in computer science and engineering from the University of Michigan-Ann Arbor, summa cum laude. She received the PhD degree from Harvard University, advised by Professor David Parkes. She is currently a software engineer at Google. Previously, she held positions as an applied researcher at Microsoft and a postdoctoral associate at Yale University, supported by a US National Science Foundation (NSF)-CRA Computing Innovations Fellowship. She received an NSF Graduate Research Fellowship and an AT&T Labs Fellowship.

Henry Lam received the PhD degree in statistics from Harvard University. He has been an assistant professor in the Department of Mathematics and Statistics at Boston University, since 2011. His research interests include large-scale stochastic simulation, rare-event analysis and simulation optimization, with application interests in service systems and risk management. His works have received funding from the US National Science Foundation (NSF) and National Security Agency (NSA), Honorable Mention Prize in INFORMS George Nicholson Best Student Paper Competition, and Finalist in INFORMS JFIG Best Paper Competition.

Zhenming Liu received the PhD degree in theory of computation at Harvard in 2012. He is a postdoctoral research associate at Princeton University, working with Jennifer Rexford, Mung Chiang, and Vincent Poor. His doctoral research lies in the intersections among applied probability, combinatorial optimization, and machine learning. More recently, he works with networking and data mining researchers to understand how these theoretical tools can help in building scalable systems for analyzing massive data sets. He has received several awards for his research, including the Best Student Paper Award at ECML/PKDD 2010.

Felix Ming Fai Wong received the BEng degree in computer engineering from the Chinese University of Hong Kong in 2007, and the MSc degree in computer science from the University of Toronto in 2009. He is currently working toward the PhD degree in electrical engineering at Princeton University.