Acting the Same Differently: A Cross-Course Comparison of User Behavior in MOOCs

Ben Gelman
Dept. of Computer Science
George Mason University
bgelman@gmu.edu

Matt Revelle
Dept. of Computer Science
George Mason University
revelle@cs.gmu.edu

Carlotta Domeniconi
Dept. of Computer Science
George Mason University
carlotta@cs.gmu.edu

Aditya Johri
Dept. of Computer Science
George Mason University
johri@gmu.edu

Kalyan Veeramachaneni
CSAIL,MIT
Cambridge, MA, USA
kalyan@csail.mit.edu

ABSTRACT
Recent studies of MOOCs demonstrate their ability to reach a large number of users, but also caution against the high rate of dropout. Some have looked closely at MOOC participation in order to better understand how and when users start to disengage, and, if they remain engaged, in what activities they participate. Most of this prior work relies heavily on descriptive statistics or clustering methodologies to highlight basic user participation characteristics. In this paper, we adapt NMF to provide a multi-dimensional view of user participation. We use log data to create a bottom-up understanding of user participation, and identify five basic behaviors associated with participants' use of content and their engagement with assessment. Furthermore, we do a cross-course analysis across four courses and find that these five behaviors are present in all courses. Interestingly, users' participation patterns - how they engage in these five behaviors - vary across courses even when the course topics are similar. Our methodology can be applied to other data sets, and findings from this work can assist in interventions to help users successfully accomplish their learning goals.

Keywords
MOOCs, Participant Behavior, NMF, Comparative Analysis

1. INTRODUCTION
As Massive Open Online Courses (MOOCs) grow in popularity, and offer an increasing variety of subjects across multiple platforms, there has been significant interest in MOOC users' participation patterns. Extremely low user completion rates [6] have motivated examinations and studies of MOOC behavior that aim to ascertain whether changes in pedagogy can improve completion outcomes, or if every incoming class contains a cohort of users that had no intention to complete.

We were motivated by this recent work to attempt to better understand MOOC users' behavioral patterns, and the evolution of participation over time and across courses. In this paper, we analyze data from four MOOC courses across three axes (*learners*, *time*, and *courses*), choosing methods that link behaviors and patterns across these three dimensions. Utilizing the rich features developed to characterize learners' weekly interactions, we adapt non-negative matrix factorization (NMF) [5] to study the importance of these features and the behavior of users over time [2].

Several factors make NMF particularly well-suited for this type of analysis. The non-negativity constraint helps to identify distinct but additive latent factors. In other words, we are able to learn user behaviors in terms of evolving parts due to NMF's additive latent factors and our temporal adaptation (linking behaviors across weeks).Through this study, we make the following unique contributions: 1) We identify behavioral patterns of users that are consistent across multiple MOOCs; 2) We demonstrate how these behaviors vary across different courses; and 3) We demonstrate the feasibility of a framework that can be applied across similar multi-dimensional datasets.

2. RELATED WORK
Several studies of MOOCs highlight low completion rates [13]. The University of Edinburgh launched six MOOCs on the Coursera platform in January 2013 [7]. Evaluations revealed that, of the 309,682 learners initially enrolled, 123,816 (about 40%) accessed the course sites during the first week ('active learners'), and 90,120 (about 29%) engaged with course content. Over the duration of the course, the number of active participants rose to 165,158 (53%). As a gauge of persistence, 36,266 learners (nearly 12%) engaged with week 5 assessments. This represented 29% of initial active learners (although individual numbers for each of the six courses ranged from 7% to 59%). In addition, 34,850 people (roughly 11% of those who enrolled) achieved a statement of accomplishment for reaching a percentage-based benchmark of course completion.

Similarly, when Duke University ran a Bioelectricity MOOC in 2012 [15], 12,175 students initially registered. Only 313 participants (2.6%) achieved a statement of accomplishment. Learner feedback suggested three specific reasons for failure to complete [15]. [8] provides a compilation of available data on MOOC completion. Further analysis of the data shows that, of the 61 courses hosted by Coursera, the average completion rate was just over 6%. This combination of MOOCs' enormous popularity and extremely low completion rate has attracted significant interest.

[17] used a classification method that identifies a small number of longitudinal engagement trajectories in MOOCs. This classifier consistently identifies four prototypical trajectories of engagement: (1) *Completing*, (2) *Auditing*, (3) *Disengaging*, (4) *Sampling*. To decide these engagement patterns, the authors used a number of *binary* variables to determine whether a student accessed a resource or attempted a problem. In contrast, we begin to extract a number of richer descriptors about the students' interaction with the online learning platform.

[9] divides participants into five profiles: no-shows (those who register but never log in); observers (those who log in but do not take assessments); drop-ins (those who participatebut do not attempt to complete the entire course); passive (those seeing the course as content to consume); and active (those participating in all the activities and enriching the course). Similarly, [16] distinguishes five groups of people depending on their level of participation in the MOOC forum: inactive (those that do not visit the forum); passive (those that just consume information); reacting (those that add further aspects to existing questions); acting (those that post questions and lead discussions); and supervising/supporting (those that lead discussions and summarize gained insights).

3. DATA

Our study utilizes four courses, including 6.002x (Fall 2012 and Spring 2013): Circuits and Electronics, 2.01x (Spring 2013): Elements of Structures, 3.091x (Spring 2013): Introduction to Solid State Chemistry. After filtering out learners who had no browsing events for the duration of the courses, the course sizes are 17379, 6339, 5597 and 8870 users, respectively. The course durations are all set to 14 weeks. Using the scripts from the MOOCdb project, we are able to extract 21 features. Table 1 shows the feature numbers and descriptions.

Figure 1 presents the course sizes dynamically. The count of active users for any week is given by the sum of users that have at least one non-zero feature in that week. The count of inactive users is the sum of users that have all-zero feature values in the current week, but had been active in a prior week. New users are those whose first non-zero feature is in the current week. The dropout value is the number of students who are inactive this week and will be inactive for all future weeks.

Because some features are complex and not fully explained by their feature names, we will expand their definitions here. Each feature is computed using the data collected in a week, and generates a single value, so if there are 14 weeks in a course, a user's feature vector will contain 14 values per feature.

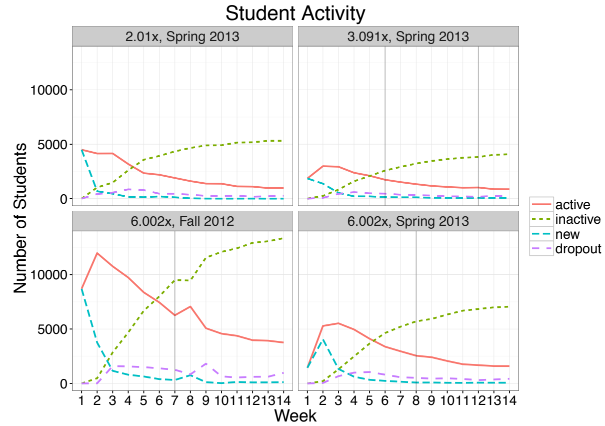


Figure 1: Student activity statuses over time for each class. Vertical lines denote midterm exams and quizzes.

Time spent: Feature 1 sums a user's total time spent on any and all events in the course. Feature 11 is the single longest time spent on any single resource (book, wiki, lecture videos, etc). Feature 12 is the time specifically spent on lectures, and feature 13 is the time spent on the course wiki.

Homework participation: Feature 4 is the count of all unique problems a learner attempted [1]. Feature 5 is the count of all attempts, including multiple tries at the same problem. Feature 6 is the count of all problems that the learner got correct (grade 1). Feature 7 is the average number of attempts per problem. Feature 18 counts all correct attempts, in order to identify users that correctly solve the same problem multiple times.

Ratio-based features: Feature 8 measures the total time spent on the course per correct problem by dividing features 1 and 6. Feature 9 divides the number of attempts (feature 5) by the number of correct problems (feature 6). Feature 19 divides total attempts (feature 5) by non-distinct correct attempts (feature 18).

Difference-based features: Features 14-17 represent the change in features 2, 7, 8, and 9, respectively. This is computed by taking the respective feature's value for the current week, subtracting the previous week, and then normalizing the result.

Regularity and procrastination: Feature 10 tells us how spread out a student's schedule is over the week by presenting the variance of his or her event timestamps. Feature 20 computes the average amount of time the user submits before the deadline (a zero value means an on-time submission, while a higher value means the word was submitted earlier). Finally, feature 21 calculates the standard deviation in working hours throughout the day—if the student starts work around the same time every day, the feature value will be low.

Feature extraction allows us to represent learners as a set of multiple time series. A learner's basic actions are collected and summarized into the 21 interpretive features on a weekly

Table 1: Students' features.

	Features' Names
1	sum_observed_events_duration
2	number_of_forum_posts
3	average_length_of_forum_posts
4	distinct_attempts
5	number_of_attempts
6	distinct_problems_correct
7	average_number_of_attempts
8	sum_observed_events_duration_per_correct_problem
9	number_problem_attempted_per_correct_problem
10	observed_event_timestamp_variance
11	max_duration_resources
12	sum_observed_events_lecture
13	sum_observed_events_wiki
14	difference_feature_2
15	difference_feature_7
16	difference_feature_8
17	difference_feature_9
18	attempts_correct
19	percent_correct_submissions
20	average_predeadline_submission_time
21	std_hours_working

basis. Because learners are represented as a set of features with per-week, aggregate values, time is a dimension of our data set.

4. METHODOLOGY

Uncovering the behaviors of MOOC students requires simultaneously finding interaction patterns (behaviors) across a large number of students and permitting individual students to exhibit multiple behaviors. Since we assume student interactions may be the result of multiple behaviors, we choose to use a decomposition method (NMF) which results in a parts-based representation of student interactions. Students may exhibit multiple behaviors and their behaviors may change over time.

Step 1: Apply NMF Given a three dimensional vector representation of the student feature data with w weeks, f features, and n users, we construct the tensor A_{ijk}. We begin by applying non-negative matrix factorization to each feature-user matrix A_i for $i = [1...w]$. We use a standard implementation [14] with NNDSVD [3] for initialization of the basis matrix and Frobenius cost function. The rank parameter, r, is set to six, which is selected through approximation.

$$\mathbf{A_i} = \mathbb{B}_i \mathbb{C}_i \tag{1}$$

The results of factorizing A_i are B_i and C_i, the *basis* and *coefficient* matrices, respectively. The dimensions of B_i are $f \times r$ and the dimensions of C_i are $r \times n$.

Each of the r column vectors in B_i contain f values that essentially describe the importance of each feature to the given column vector. In our data, we use the set of important features in each basis vector to describe a behavior. In matrix $C_i^\intercal$, there are r column vectors that contain n coefficient values, one for each user. The m^{th} column vector's coefficient values in $C_i^\intercal$ describes how closely users associate with the m^{th} basis vector in B_i. Because every user has r coefficient values, it is possible for a user to identify with multiple basis vectors. This is significantly different than hard clustering approaches such as K-means, where groups are mutually exclusive.

Step 2: Alignment After performing the matrix factorization on each week, we have w basis matrices and w coefficient matrices. To identify persistent basis vectors and patterns, we must connect the results over time. There is no guarantee the order of the basis vectors is consistent over all weeks because the basis matrices are produced by independent executions of NMF. To achieve this, we first compute the cosine similarity using Equation (2) between two consecutive basis vectors. In other words, for each of the r basis vectors in week i, we compute the cosine similarity to all basis vectors in week $i+1$, resulting in r^2 computations. Ultimately, there are $(w-1)r^2$ similarity computations.[1]

$$\text{Sim}(u, v) = \frac{u \cdot v}{||u||_2 ||v||_2} \tag{2}$$

By examining the distribution of cosine similarity values, an alignment threshold may be selected. For our data, a threshold value of 0.95 was chosen to identify matching basis vectors between weeks. We found that after the first week, all basis vectors uniquely match *one and only one* basis vector in the consecutive week when a threshold of ≥ 0.95 is used. This phenomenon occurred for all four courses we used in our experiments. Although basis matrices for each week are estimated independently, we find five basis vectors which persist over time and occur in all the classes.

Step 3: Normalize and define behaviors The aligned, per-week basis vectors are normalized. We then average these aligned-normalized vectors into a single, representative *behavioral* vector. Having a single, normalized vector permits a semantic interpretation of the behavior based on relative feature values. By identifying the most important features (the ones with the largest values) in each *behavioral* vector, we are able to label the vectors by the interaction pattern they best represent.

Step 4: Coefficient analysis

Every student's interaction attributes may be approximated using a weighted mixture of the discovered behavior vectors. These weights (coefficients) can be considered to define a soft-membership of a student to a behavior.

In order to decide if a user belongs to a behavior, we threshold the distribution of the coefficient values per

[1]We choose cosine similarity because it is a measure of angular similarity between two vectors. Thus, two basis vectors whose only nonzero entry is feature j will be extremely similar. This is valuable for aligning basis vectors whose distributions of features are similar.

week and per behavioral vector (or basis). This means that the algorithm will generate $r \times w$ thresholds. The thresholding algorithm takes the entire range of coefficient values per vector and limits the range of values to the top $x\%$. The threshold (top $x\%$) is a parameter. This means that if the range of coefficient values for a behavior is 0-100, then selecting a threshold of 0.85 will only consider users with coefficient values of 85-100 to be exhibiting that behavior. There is an additional minimum size parameter s that adjusts for a skewed distribution where a few users have significantly higher coefficient values that any other users. This skewed distribution causes the top $x\%$ of coefficient values to only include these few users. If the number of users within the top $x\%$ is less than the s, then the users will be saved, and the threshold computation will be repeated without them. For our data, we use a threshold of 0.85 with a minimum size parameter of 30.

We assign behaviors to students for each week using the data-derived thresholds. By tracking the set of behaviors across weeks, we generate a transition diagram that presents the number of students exhibiting each behavior over each week and the migration of users between various behaviors. The transition diagram allows us to understand the evolution of user behavior as a course progresses.

Table 2: Matrix of normalized basis vectors (behaviors) for week 1 (course 6.002x fall 2012). The behaviors *Introduction* and *Sampling* are unique to week 1. Dominant feature values are shown in boldface.

Feature	*Deep*	*Consistent*	*Bursty*	*Introduction*	*Sampling*
1	0.012	0.000	0.001	0.000	0.088
2	0.000	0.000	0.000	**0.137**	0.000
3	0.000	0.000	0.000	**0.862**	0.000
4	0.000	0.000	0.000	0.000	0.000
5	0.000	0.000	0.000	0.000	0.000
6	0.000	0.000	0.000	0.000	0.000
7	0.000	0.000	0.000	0.000	0.000
8	0.000	0.000	0.000	0.000	0.000
9	0.000	0.000	0.000	0.000	0.000
10	0.000	**0.988**	0.000	0.000	0.000
11	**0.981**	0.011	0.000	0.001	0.000
12	0.000	0.000	0.000	0.000	**0.665**
13	0.000	0.000	0.000	0.000	0.000
14	0.000	0.000	0.000	0.000	0.000
15	0.000	0.000	0.000	0.000	0.000
16	0.000	0.000	0.000	0.000	0.000
17	0.000	0.000	0.000	0.000	0.000
18	0.000	0.000	0.000	0.000	0.000
19	0.000	0.000	0.000	0.000	0.000
20	0.000	0.000	0.000	0.000	0.000
21	0.008	0.000	**0.999**	0.000	**0.248**

5. BASIS MATRIX RESULTS

The resulting basis matrices for 6.002x (Fall 2012) exhibit eight unique behaviors. Tables 2 and 3 numerically summarize behaviors for week one and the average of the other weeks, respectively. Because the first week manifests two unique behaviors, namely *introduction* and *sampling*, it is kept separate. From the second week onwards, all behaviors are persistent (at least 95% cosine similarity). This allows us to average weeks two through 14 in Table 3.

Basis vector one is dominated by feature 11 (max_duration_resources), which is the duration of the longest observed event this week. This vector represents a *deep* behavior, because the associated students must have spent a long time on a single resource.

Basis vector two is primarily decided by feature 10 (observed_event_timestamp_variance). Because this feature tells us how spread out the student's schedule is over the week, this vector describes a *consistent* behavior. Having a high timestamp variance requires users to log in multiple times a week.

Basis vector three is primarily decided by feature 21 (std_hours_working), which is the standard deviation in working hours over the day. This could represent a *bursty* behavior—because a user must be active during different times in a day to obtain a high feature value, this could mean that the user has a single prolonged session or multiple, separate sessions.

Two basis vectors exist only in the first week of the course. Basis vector four in Table 2 is decided by feature three (average_length_of_form_posts) and feature two (number_of_form_posts). This supports the idea that users interacted heavily during the opening week of the course. The disappearance of this basis vector, however, tells us that forum interaction in later parts of the course was insignificant in 6.002x fall 2012. For this reason, this basis vector characterizes an *introduction* behavior.

Basis vector five in Table 2 is defined by features 12 (sum_observed_events_lecture), 21 (std_hours_working), and 1 (sum_observed_events_duration). This group of features supports the hypothesis that users are browsing through a lot of content during the first week of the course. This may be because users are interested in seeing what lies ahead in the course, or because some users may have joined only to gather information on one particular topic. Thus, basis vector five during the first week expresses a *probing* behavior.

After the first week, two more basis vectors persist. At this point, basis vector four is primarily characterized by feature 19 (percent_correct_submissions). By turning in assignments with high correctness, the corresponding students can be associated with a *performance* behavior. Basis vector five is strongly defined by feature 20 (average_predeadline_submission_time). By turning in assignments long before their deadlines, these students can be associated with an *response* behavior.

When we apply the same analysis to other courses, we see similar behaviors. The average basis matrix tables for 2.01x, 3.091x, and 6.002x are not displayed because they exhibit the same behaviors as table 3 with 95% cosine similarity. It appears that each of these five behaviors— deep, consistent, bursty, performance, and response—appear in all of the courses. The key difference is that 6.002x has two additional behaviors that occur only in the first week. The introduction and sampling behaviors do not appear to be prevalent in the other courses. This could be due to course

Table 3: Average matrix of normalized basis vectors for weeks 2 through 14 (Course 6.002x, Fall 2012). Dominant feature values are shown in boldface.

Feature	*Deep*	*Consistent*	*Bursty*	*Performance*	*Response*
1	0.031	0.002	0.007	0.000	0.000
2	0.001	0.000	0.001	0.000	0.000
3	0.004	0.001	0.003	0.000	0.000
4	0.005	0.000	0.000	0.000	0.029
5	0.003	0.000	0.000	0.001	0.012
6	0.000	0.000	0.000	0.052	0.000
7	0.001	0.000	0.000	0.003	0.003
8	0.000	0.000	0.000	0.001	0.001
9	0.000	0.000	0.000	0.001	0.001
10	0.001	**0.993**	0.000	0.000	0.000
11	**0.922**	0.000	0.005	0.007	0.028
12	0.010	0.000	0.002	0.000	0.000
13	0.000	0.000	0.000	0.000	0.000
14	0.001	0.000	0.000	0.000	0.000
15	0.001	0.001	0.000	0.002	0.002
16	0.000	0.000	0.000	0.000	0.000
17	0.000	0.000	0.000	0.000	0.000
18	0.000	0.000	0.000	0.015	0.000
19	0.002	0.000	0.000	**0.743**	0.004
20	0.000	0.000	0.000	0.174	**0.920**
21	0.017	.0000	**0.980**	0.000	0.000

sizes, and the fact that 6.002x was the first edX course ever released. Users may have been encouraged to communicate in the forums early on (introduction), or there may have been users testing the waters of this new online course platform (sampling).

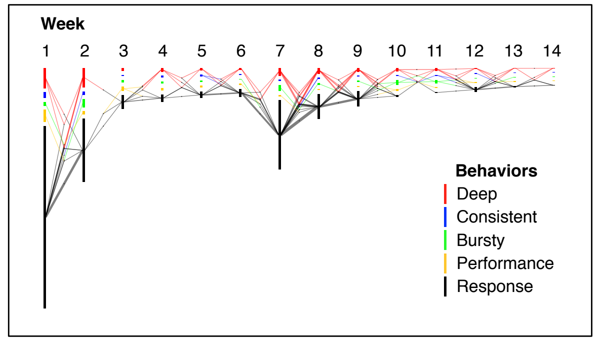
6. STUDENT TRANSITIONS

After applying the thresholding algorithm, we generate user behavior transition diagrams for each course. The size of each colored bar is scaled according to the amount of students exhibiting the behavior. The transition lines in between the bars are sized and directed based on user migration between sets of behaviors.

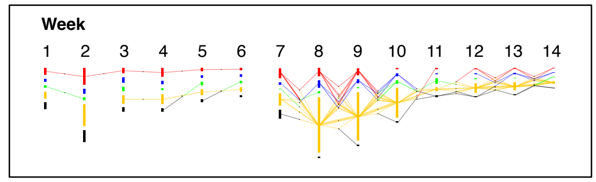
Using these diagrams, we can observe changes in the behaviors themselves, and the transitional motifs that occur due to user migration. After the first week or two, a single behavior persists as the largest. Additionally, this behavior tends to act as a hub for user migration. This phenomenon significantly highlights the fact that the behaviors may manifest differently despite the existence of the same five behaviors among all five courses.

In 2.01x, most user migration occurs into and out of the response behavior, with a secondary focus on the deep behavior. Notable moments occur in week 5 and weeks 10 to 12, where migration between consistent and deep occur. Otherwise, there are several recurrent transitions. These motifs include each permutation of deep and/or response migrating to deep and/or response.

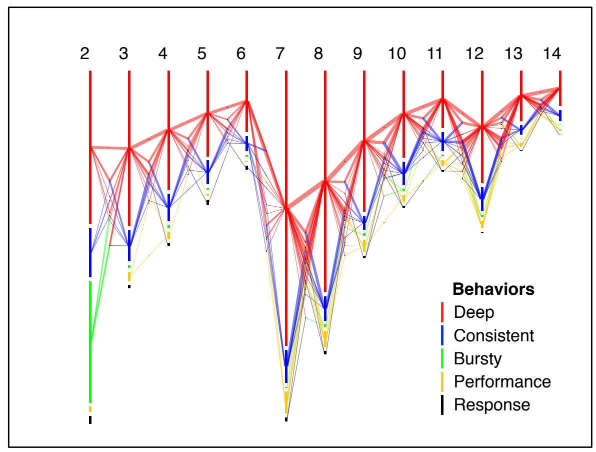
In 3.091x, most user migration occurs into and out of the performance behavior. Most unusually, there is very little migration in the entire first half of the course. Only in the second half does migration pick up to levels we would have expected given the results of the other courses. Although some migration patterns through the performance behavior



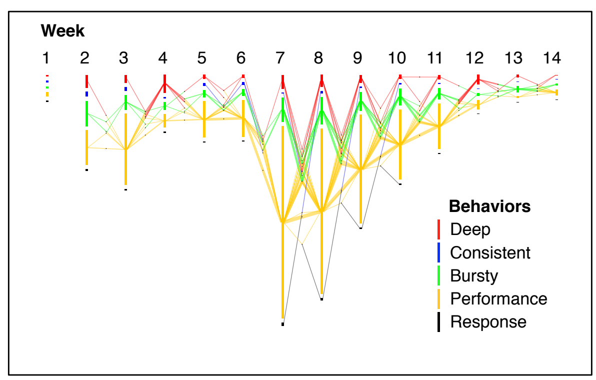
(a) 2.01x, Spring 2013



(b) 3.091x, Spring 2013



(c) 6.002x, Fall 2012



(d) 6.002x, Spring 2013

Figure 2: User behavior transitions over time. Vertical bars are numbers of students performing each behavior. Diagonal groups indicate transitions: for example, the transition indicates students who were Deep and Bursty and have transitioned to Consistent. Transition thickness is the log of the number of students involved.

repeat occasionally, they only occur for two to three weeks at a time. Thus, we do not infer any transitional motifs from this course.

In 6.002x fall, most user migration occurs through the deep behavior, with a secondary focus on the consistent behavior. A unique circumstance occurs between weeks one and two with the migration of the initially enormous bursty behavior. Besides this, the transitional motifs include each permutation of deep and/or consistent migrating to deep and/or consistent.

In 6.002x spring, most user migration occurs through the performance behavior. Unlike the other courses, there are two more behaviors through which there is significant migration: the deep and bursty behaviors. As a result, we see many more motifs than simply the permutations of the top two behaviors. In the early weeks, migration is heaviest through deep and performance. This means that early on, users are both engaged and performing well. In the middle weeks, during and after the midterm, there is a chaotic shuffle between behaviors as users deal with the course differently. In the later weeks, however, deep migration falls off and users mostly move between bursty and performance. This may suggest that users are capable of finishing their work in a single day or two and achieving high correctness simultaneously. This result could perhaps reflect a decreased difficulty in the later weeks of the course. The occurrence of multiple large behaviors appears to tells us more about the evolution of user behavior.

7. CONCLUSION

In this comparative study of four MOOC courses, we show how users follow five specific behaviors across the courses. We found that although these behaviors are common, their patterns of occurrence vary across courses. Through our multi-dimensional data and our adaptation of NMF, the results reveal in great detail the differences in behavior over time between the courses. Because our method analyzes behavior at every step of the MOOC experience, our work can improve the learning experience for all users, not just those that plan to finish the course. For future work, we can expand the purposes of user behavior trajectories by using Markov modeling for prediction. We can add newer, more descriptive features in addition to running the analysis with a higher rank in order to discover possible alternative behaviors. If course outcomes and assessment information are available, we can combine these with the dynamic behavioral motifs to better understand the underlying processes that fuel behavioral changes.

8. REFERENCES

[1] Veeramachaneni, Kalyan, Halawa, Sherif, Dernoncourt, Franck, O'Reilly, Una-May, Taylor, Colin, and Do, Chuong. Moocdb: Developing standards and systems to support mooc data science. arXiv preprint arXiv:1406.2015, 2014a.

[2] Sra, Suvrit, and Inderjit S. Dhillon. "Generalized nonnegative matrix approximations with Bregman divergences." Advances in neural information processing systems. 2005.

[3] Boutsidis, Christos, and Efstratios Gallopoulos. "SVD based initialization: A head start for nonnegative matrix factorization." Pattern Recognition 41.4 (2008): 1350-1362.

[4] Swinson, Christina J. "Mathematica." CJs Blog of Miscellanies and Accelerator Physics. N.p., 21 Jan. 2011. Web. 05 Mar. 2016.

[5] Lee, Daniel D., and H. Sebastian Seung. "Learning the parts of objects by non-negative matrix factorization." Nature 401.6755 (1999): 788-791.

[6] Breslow, Lori, et al. "Studying learning in the worldwide classroom: Research into edX's first MOOC." Research & Practice in Assessment 8 (2013).

[7] MOOCs@Edinburgh Group. MOOCs@Edinburgh (2013): Report#1, Available at: http://hdl.handle.net/1842/6683 [Accessed: 20/01/14].

[8] Jordan, K. (2013). MOOC Completion Rates: The Data, Available at: http://www.katyjordan.com/MOOCproject.html [Accessed: 18/02/14].

[9] Hill, P. (2013). The Most Thorough Summary (to date) of MOOC Completion Rates|e-Literate. e-Literate blog. Retrieved June 10, 2013, from http://mfeldstein.com/the-most-thorough-summary-to-date-of-mooc-completion-rates/

[10] Liyanagunawardena, T. R., Adams, A. A. & Williams, S. A. (2013). MOOCs: a systematic study of the published literature 2008–2012. The International Review of Research in Open and Distance Learning, 14, 3, 202–227.

[11] Ebben, M. & Murphy, J. S. (2014). Unpacking MOOC scholarly discourse: a review of nascent MOOC scholarship. Learning, Media and Technology, 39, 3, 1–18. doi: 10.1080/17439884.2013.878352

[12] EDUCAUSE (2012). What Campus Leaders Need to Know About MOOCs. EDUCAUSE BRIEFS. Retrieved June 10, 2013, from http://www.educause.edu/library/resources/what-campus-leaders-need-know-about-moocs

[13] Onah, Daniel F. O., Sinclair, Jane and Boyatt, Russell (2014) Dropout rates of massive open online courses : behavioural patterns. In: 6th International Conference on

[14] Pedregosa et al., Scikit-learn: Machine Learning in Python, JMLR 12, pp. 2825-2830, 2011. Education and New Learning Technologies, Barcelona, Spain, 7-9 Jul 2014. Published in: EDULEARN14 Proceedings pp. 5825-5834.

[15] Belanger, Y. (2013). Bioelectricity : A Quantitative Approach, Available at http://dukespace.lib.duke.edu/dspace/bitstream/handle/10161/6216/Duke_Bioelectricity_MOO C_Fall2012.pdf

[16] F. Gruenwald, E. Mazandarani, C. Meinel, R. Teusner, M. Totschnig, and C. Willems, "openHPI-a Case-Study on the emergence of two learning communities," in Proc. IEEE Global Eng. Edu. Conf., 2013, pp. 13–15.

[17] Kizilcec, R. F., Piech, C., & Schneider, E. (2013, April). Deconstructing disengagement: analyzing learner subpopulations in massive open online courses. In Proceedings of the third international conference on learning analytics and knowledge (pp. 170-179). ACM.