

MONITORING A MOOC: THE ANALYTICS MODULE OF A MOOC MANAGEMENT SYSTEM

Ravi Mula, Rajat Jain, T V Prabhakar
Department of Computer Science, IIT Kanpur, UP 208016, India
{rvsmula, rajatjn, tvp}@iitk.ac.in

Abstract

Online courses offer interesting opportunities to observe student behavior and learning patterns. All the interactions of the student from login to logout are recorded. A proper analysis of this data can reveal the engagement level of the student with the course, with the course content and with other learners. In the absence of face-to-face-interactions with students, this is an invaluable feedback to the Instructor. mooKIT is a versatile MOOC Management System designed with a state-of-the-art learning analytics. mooKIT logs all events that happen during a user session and provides a powerful graphical user interface for the Instructor to make sense out of this. The analytics are presented at a lecture level (like what is the viewing pattern of a lecture), and at an individual student level, on a time scale. These can be used to figure out the difficulty level of each lecture and for each student. Analysis of the Forum activities, like volume of questions, interaction among students, the sentiment of the comments etc. can be used to get a good understanding of the social behavior and comfort level of the students. By being able to predict the students who are likely to drop out from the course, the system enables the Instructor to take remedial actions. One can use the predictive analytics to configure the machine for hosting. In this paper, we discuss the kind of learning analytics that are possible using mooKIT as an example.

1. INTRODUCTION

"Quality without Exclusivity"- that is what MOOCs (Massive Open Online Courses) have done to education. Anybody, from anywhere, at any time, is able to listen to the best teachers in the world, making it 'flat', as Thomas Friedman (World is Flat, Book, 2005) would say. The New York Times labelled 2012 as the *Year of the MOOC* and by now MOOCs have entered the vocabulary of every aspiring student. MOOCs currently offer about 4500 courses running worldwide from various universities^[1]. With every course in a MOOC, comes a huge amount of unstructured data about student activity. This data involves logging of nearly everything a student does from logging in to logging out. This data has a huge potential to uncover various interesting patterns in the activities of the students which can be useful for an instructor to plan course strategies. Therefore, there's a need to analyze it and extract useful inferences from it.

To analyze and monitor the data of mooKIT^[2], we have built a state-of-the-art learning analytics module. This module provides a comprehensive analysis of student activity. It provides the Instructor an easy-to-use, self-explanatory dashboard to observe the behavior patterns at an aggregated level as well as at an individual level. In this paper, we provide the details of this platform along with some visuals.

1.1. mooKIT

mooKIT is a lightweight MOOC Management System conceived, designed and developed at the Indian Institute of Technology, Kanpur targeted towards developing nations. It is aimed at embodying the following principles:

- Content first, platform next - easy and intuitive to use.
- Course creation and management should be simple.
- Multi-platform (supports Mobiles, Tablets, etc.)
- Efficient with server resources.
- Built on open source technologies.

1.2. Architecture of mooKIT

mooKIT follows a REST Architectural style. It uses the Drupal^[3] Content Management System as its backend, which takes care of the user and content management. The core engine runs on the MEAN stack, making it extremely scalable and can operate on low end servers, supporting a large number of simultaneous users. The student interface is really simple and intuitive, built with internet novices in mind. The architecture of mooKIT is shown in Fig 1.

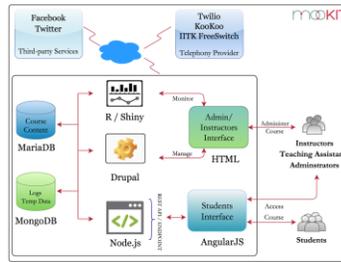


Fig 1: Architecture of mooKIT

1.3. Features of mooKIT

- *Scalability*: Rendering the HTML of the page on the server incurs a cost resulting in wastage of its resources. By shifting the load to the client, the server is able to respond faster. This results in improved performance, i.e. the server is capable to handle a large number of simultaneous users in parallel, thereby cutting down the hosting costs.
- *Customizability*: As mooKIT has Drupal as a part, it takes care of Database schema. This helps the developers to add new functionalities and integrate them with JavaScript API (Node.js) quickly.
- *Low Bandwidth Scenarios*: mooKIT has unique features addressing low bandwidth scenarios. It constantly provides feedback indicating the status of the user's connection. Users can choose the mode of delivery based on their bandwidth with the options being Video, Audio and Phone call.
- *Hangouts* allow the users to communicate with co-learners and instructors in real time. Instructors can choose an option of going live to interact with students and answer their queries either on a video or by text.
- *Integration with Social networking sites* allows the users to access the discussions on the platform from Facebook and Twitter. This allows the users to keep a track of discussions without visiting the platform.
- *Forums* allows the users to participate in discussions, clear their doubts, interact with co-learners and share thoughts etc.

1.4. Need for Analytics

In MOOCs, face-to-face interaction between a student and an instructor is not possible, in which case the Instructor needs to know the students' response/feedback to the provided material. Learning Analytics help Instructors at various stages during the course starting from planning till evaluation. Detailed monitoring of each activity of the students results in gaining insights like "What is the time when most students are active?", "How difficult / easy was the lecture?", "Do I need to provide additional material for a lecture to help the students better understand the topics?" and many more interesting patterns that were not possible in a face-to-face teaching. In mooKIT, we categorize, compute and show the Instructor an aggregated as well as individual metrics to understand the behavior of the students.

2. THE ANALYTICS MODULE OF mooKIT

mooKIT has an easy and intuitive module dedicated to monitor the student participation. The analysis is classified into 3 categories based on the phases involved during the course.

- Registration*: The data collected during the registration results in various insights which the Instructor can use for planning the course. It can be used to check the profile of the registered students. One gets an insight on day-to-day registrations, worldwide distribution of the audience and more.
- Interactions*: After registration, students go through the content, participate in discussions and share thoughts etc. An aggregate analysis of the activities of students during interaction gives an understanding about their behavior like how frequently they login, interact in forums and much more.
- Evaluation*: Students get certified on the basis of their performance in the assessments. An analysis on the performance would help in figuring out how well the content was received by the students, difficulty level of the questions and to identify dropouts etc.

For each of these phases a set of graphs are presented for easy understanding of the data. To collect the data, mooKIT tracks each and every click made by the user. A sample of the click data is shown in Fig 2 which consist of the user ID, the click time, type of request, URL of the resource, IP address from which the request was made, etc. Apart from the clicks, based on the type of the event, some more data is collected.

```
{ "_id" : ObjectId("574b3c22f854ea7e0bb15fec"), "uid" : 1, "timestamp" : 1464548387,
  "ip" : "172.20.3.47", "method" : "GET", "URL" : "/lectures/summary", "from" : "Web"
}
{ "_id" : ObjectId("574b3c22f854ea7e0bb15fed"), "uid" : 1, "timestamp" : 1464548387,
  "ip" : "172.20.3.47", "method" : "GET", "URL" : "/assignments/summary", "from" : "Web"
}
{ "_id" : ObjectId("574b3c22f854ea7e0bb15fee"), "uid" : 1, "timestamp" : 1464548387,
  "ip" : "172.20.3.47", "method" : "GET", "URL" : "/quiz/summary", "from" : "Web"
}
{ "_id" : ObjectId("574b3c23f854ea7e0bb15fef"), "uid" : 1, "timestamp" : 1464548387,
  "ip" : "172.20.3.47", "method" : "GET", "URL" : "/announcement/urgent", "from" : "Web"
}
```

Fig 2: Sample data stored for a click

2.1. Registration

When students register, some personal information is collected for Identity Management. Every course is targeted towards a specific audience and may have some prerequisites. The information collected for Identity Management can be used to check this. Based on this, the Instructor will also be able to provide extra material to plug in the gaps if any, in the audience.

Typically, apart from the information like Name and Email, the Age Group that the student belongs to, their Qualification, Affiliation, Location etc. are collected.

2.1.1. Daily Registrations

Here we display the number of users registered on each day [Fig 3] and on a cumulative basis [Fig 4]. This may help in planning the publicity of the course.



Fig 3: Daily Registrations



Fig 4: Total Registrations

2.1.2. Geographic Reach

In a MOOC, the students are from all over the world. Based on the worldwide distribution of the students an inference can be “If the students are from a locality where the accent is different from the Instructor’s then subtitles and/or transcripts can be provided for better understanding”. The chart is also available at country level to infer the state level participation, which can be used in cases where the course is targeted towards a specific country.

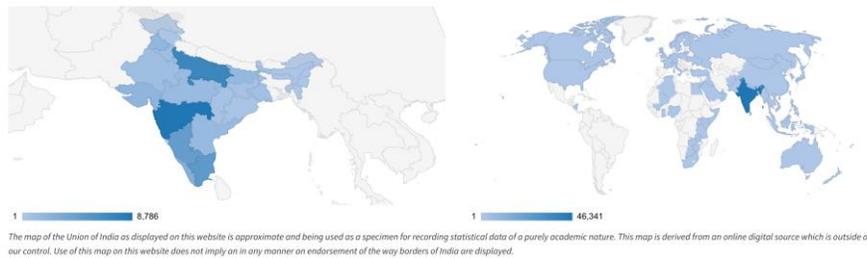


Fig 5: Country and World Wide Distribution

2.1.3. Other Information

In Fig 6 one can see the other information like the Gender, Age group, Qualification and Affiliation depicted as pie charts. Such information is useful to figure out the background and distribution of the registered students.

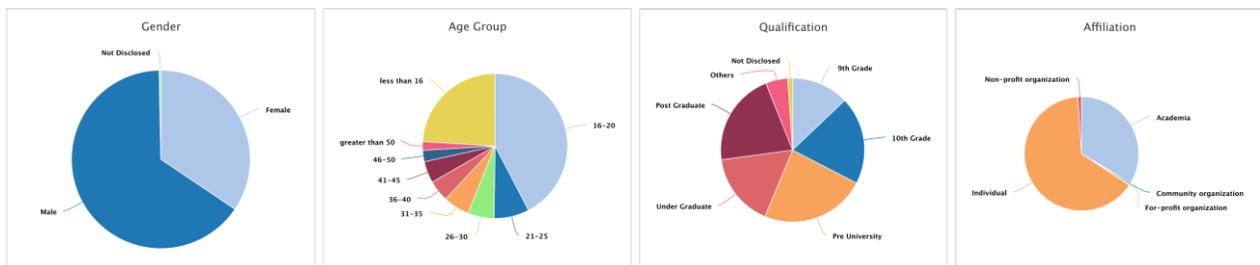


Fig 6: Gender, Age, Qualification and Affiliation Distribution

2.2. Interactions

When student’s login into the platform, they watch a video, participate in a discussion or simply browse etc. These activities are shown as charts in the Interactions subgroup of mookIT analytics.

2.2.1. Daily Engagement Summary

As a course Administrator or an Instructor, one will be interested in knowing “How many students are active on each day?”, “Did they view a lecture or participate in a discussion?” and so on. Also one can correlate the activity of students with a “Deadline for an

assessment”, an “Announcement which is sent”, the “Lecture release schedule”, the “Day/Time when the instructors are active in discussions” etc.

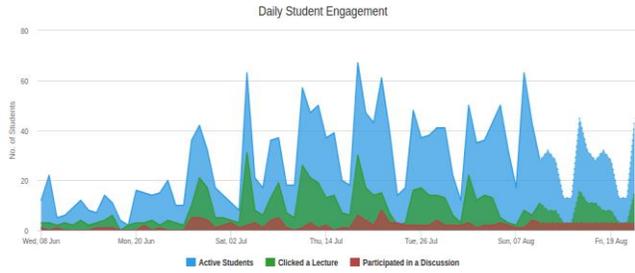


Fig 7: Daily Interaction Chart

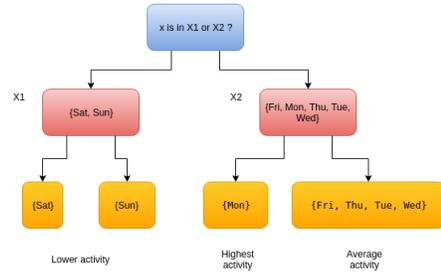


Fig 8: Decision Tree

The chart [Fig 7] also shows the engagement pattern for the coming weeks (highlighted with a dotted line). This help the course administrators to figure out questions like ‘How many students will be coming to the class next week?’. This information is useful in provisioning the hardware and other resources necessary to run the course. The prediction model observes the weekdays and their corresponding activity and generates the pattern for coming weeks with the help of Decision Trees^[4,5] [represented in Fig 8].

We can easily identify the time when the students are most active. Resources can be allocated accordingly either to the entire system or to a particular component, if the architecture permits. A sample diagram is presented in Fig 9. The activity can be one of Login / Visit / View (Complete Views of Video) / Forum.

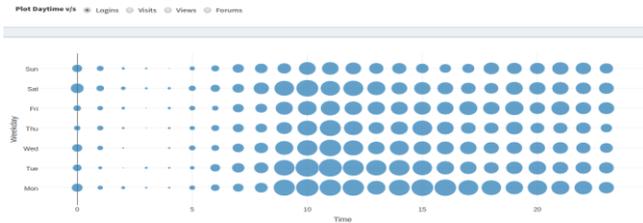


Fig 9(a): Involvement pattern for Weekdays across 24 hours

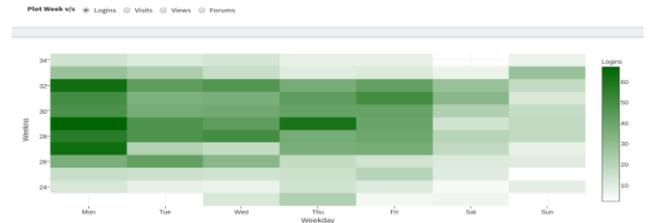


Fig 9(b): Involvement pattern for Each week across weekdays

2.2.2. Deep Video Analytics

In an online course, the usual way of delivering content is by videos. There might be content other than videos, which are additional material related to the lecture, slides and transcripts of the lecture etc. Deep analysis of the activities performed by students when they watch a lecture, lets the Instructors know how the lecture was received by students. By identifying the difficulty level of each lecture, additional material can be provided. At lecture level, “Difficult / Interesting” parts can be identified with the use of repeated and distinct views at every 10th second of the lecture shown in Fig 10(a). At lecture level, we have a knowhow on “What is the percentage of the lecture that is viewed by the students?” as shown in Fig 10(b). The difficulty index of each lecture is represented on a scale of 0 to 1 - 0 being easy and 1 being difficult and is calculated using the formula represented below.

$$\text{Difficulty Index of Lecture} = \frac{\text{No. of Repeated Views of the lecture}}{\text{MAX(No. of Repeated Views of all lectures)}}$$

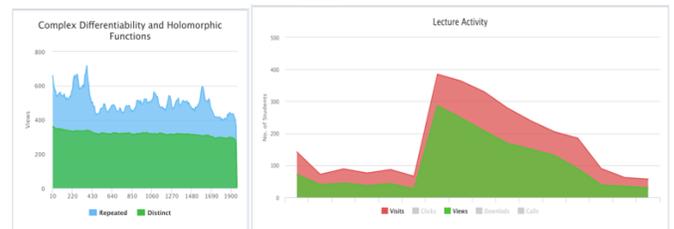


Fig 10: (a) Individual Lecture Analysis (b) Aggregated Lecture Analysis

2.2.3. Forum Analysis

Discussions are purely text based. Since students express and share their thoughts and collaborate amongst each other online, analysis of the data results in various insights. “Sentiment for each topic and for each comment [Fig 11(a)]” can be identified. Also, we can figure out how the sentiment varies each day. Further, we can identify the popular words in the forums from the Word Cloud [Fig 11(b)].

Network analysis [Fig 11(c)] results in gaining knowledge on “How the students are connected to each other?” and the way they communicate. “Are there some students who participate in most of the discussions?” and “Degree of separation between students” are some inferences that can be made from network analysis.



Fig 11: (a) Sentiment Analysis (b) Word Cloud (c) Network Graph

2.3. Evaluation Analysis

Due to the tremendous growth in online learning and its recognition by companies in recruitment, students would like to showcase their participation in the course. Hence certification is necessary. Eligibility for the certificate is decided by the performance of the student in the assessments. Based on the criteria, certificates can be of different types, like Participation, Distinction, Verified etc.

In MOOCs, as registrations are much much larger compared to a normal class, evaluations are usually done through computer gradable questions like Multiple Choice Questions. Pass/Fail marks, once decided by the Instructor, can be used to award the certificates. Performance analysis of the submissions helps us in identifying the “Difficulty level for each question”, “Weak and Strong points of students”, “Drop-outs after each assessment”, “Common students in all or any two assessments”, etc. Fig 12 & 13 are some sample representations of Evaluation Analysis.

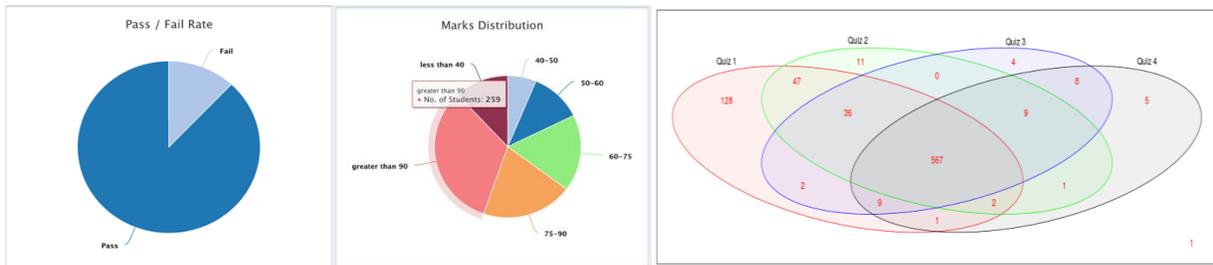


Fig 12: Marks Distribution and Relative Student Participation

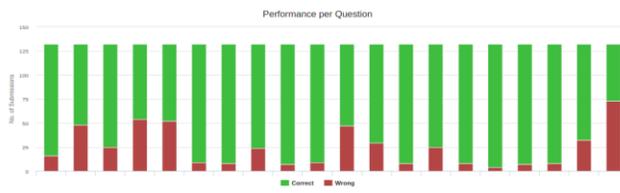


Fig 13: Performance analysis on questions of a particular quiz

2.4. Individual Analytics

In case, when a Teaching Assistant is allocated for a set of students, individual monitoring helps in understanding the behavior of each student and act accordingly to help them achieve success. Fig 14 displays statistics for each user and the grade they obtained based on the certification criteria.

TIRUMURU GANESH	55	50	85	95	95	95	60	Competency
VISWAJYOTHI K	52	40	85	90	90	95	75	Competency
ABDUL HAMEED J	47	12	100	100	100	100	65	Participation
ABHA LOHANI	55	37	80	75	85	95	20	Participation
ABRAR JAVED	55	39	85	100	85	95	75	Participation

Fig 14: Individual Student Statistics

2.5. Architecture and Implementation

The analytics module runs as a service and communicates with all the courses running on mooKIT. Periodically the data for each course is fetched, cleaned, and stored in the local storage after partial processing. MongoDB^[6] is used for local storage because of faster reads and writes. Whenever an Instructor interacts, the data is fetched from local storage, processed and displayed with the help of charts such that the Instructor can understand and gain some insights.

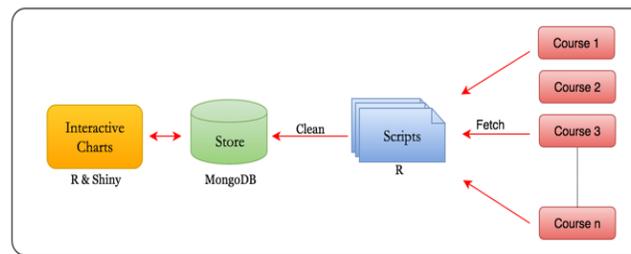


Fig 15: Architecture/Pipeline of Analytics

The base engine runs on R^[7], as it is open source and one of the popular languages used for Statistical Analysis. Also, R has a wide community of developers who work actively in bringing more functionalities. R also has integration with other tools like Spark, Watson of IBM and more. Shiny^[8] & Shiny Dashboard^[9] along with HTML Widgets^[10], rCharts^[13] and Plotly^[12] are used to provide interactivity to the end-user. Fig 15 describes the architecture of the learning analytics module of mooKIT.

3. CONCLUSIONS AND FUTURE WORK

As the number of MOOCs are increasing exponentially every year, monitoring them exhaustively and coming out with some conclusions becomes the need of the hour. The overall performance of the course can be improved by closely monitoring the activities and taking necessary actions at any point. In this paper, we have discussed and explained the kind of analysis mooKIT provides and some inferences/predictions that are possible from this data.

The future engagement prediction does not currently consider that the student engagement might have gone up due to transient events like Assessment deadlines, Announcements etc. Incorporating them could result in more precise predictions.

We would like our system to identify the students who are falling back and help them complete the course by providing additional resources if necessary. Also we would like to see how many students have accessed the additional material and compare the difference in performance between the students who have accessed the material with those who have not accessed it.

The difficulty level of the lecture is calculated based on the number repeated views. It may be possible to map the questions in the assessments to the corresponding lecture and use the performance of the students in the assessment to identify the level of difficulty of the lecture. The in-Video questions can also give some indication.

REFERENCES

1. Online Course Report, Retrieved August 14, 2016, from: <http://www.onlinecourereport.com/state-of-the-mooc-2016-a-year-of-massive-landscape-change-for-massive-open-online-courses/>
2. mooKIT <http://mookit.co>
3. Drupal, Retrieved August 20, 2016, from: <https://www.drupal.org/>
4. Padraic G. Neville (4 August 1999), Decision Trees for Predictive Modelling [https://datamining.bus.utk.edu/Documents/Decision-Trees-for-Predictive-Modeling-\(Neville\).pdf](https://datamining.bus.utk.edu/Documents/Decision-Trees-for-Predictive-Modeling-(Neville).pdf)
5. Mrinal Pandey & Vivek Kumar Sharma (13 January 2013), A Decision Tree Algorithm Pertaining to the Student Performance Analysis and Prediction <http://research.ijcaonline.org/volume61/number13/pxc3884822.pdf>
6. MongoDB, Retrieved August 12, 2016, from: <https://www.mongodb.com/>
7. R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria <https://www.r-project.org/>
8. Shiny, Retrieved August 15, 2016, from: <http://shiny.rstudio.com/>
9. Shiny Dashboard, Retrieved August 15, 2016, from: <http://rstudio.github.io/shinydashboard/>.
10. HTML Widgets, Retrieved August 15, 2016, from: <http://www.htmlwidgets.org/>.
11. Vaidyanathan Ramnath, rCharts, Retrieved August 15, 2016, from: <https://ramnathv.github.io/rCharts/>
12. Plot.ly, Retrieved August 16, 2016, from: <https://plot.ly/>