Building Learning Analysis System with GQM Methodology and ELK Stack

Nien-Lin Hsueh, Jun-Jie Wang, Daramsene Bilegjargal*

Department of Information Engineering and Computer Science, Feng Chia University, Taiwan
nlhsueh@fcu.edu.tw, j86121100@gmail.com, p0731809@o365.fcu.edu.tw

Abstract

With the development of Internet technology, Massive Open Online Courses (MOOCs) are becoming popular around the world, and more and more people are learning new knowledge through online learning platforms. Learners generate a lot of information on learning platforms every day, and most of the information is recorded in logs. However, it is difficult to start to extract valuable information from the huge and messy log data of learning platforms, which requires a methodology to guide and a lot of time and cost to process the data. Therefore, this research proposes a data analysis process, using the GQM (Goal Question Metric) method as a guide for the analysis process combined with the Banerjee analysis model to build a series of question lists and metrics to evaluate students' learning behavior performance, and using the ELK Stack (Elasticsearch Logstash Kibana) as an analysis environment to solve the problem of data processing. Finally, we conduct a case study of a programming course in the OpenEdu e-learning platform to help educators transform the log data into analyzable information to understand students' learning behaviors in the online course and to propose effective decisions for improving learning outcomes.

Keywords: Education technology, Massive Open Online Courses, Goal Question Metric

1 Introduction

As Massive Open Online Courses (MOOCs) have flourished around the world, online open learning platforms such as edX and Coursera have achieved good results, and many universities and companies are gradually using online courses to teach, and more and more people are acquiring knowledge through online learning. Compared to traditional face-to-face courses, online courses have the advantage of being independent of time and place, allowing people to learn when and where they want, which is a major factor in the growing number of people choosing to learn through online courses.

However, in the online environment it is difficult for teachers to control the learning situation of students. For educational researchers, the log data of online courses contain a lot of fine-grained information, which can be analyzed to get a lot of useful information from them.

Therefore, it has been one of the research goals of educational researchers to analyze the huge amount of data generated to understand how students learn and provide methods to improve teaching and learning [12]. The ability of ELK Stack to quickly extract data from logs and visualize them provides a simple and powerful solution for our analysis [15].

The logs generated by the MOOCs platform contain a lot of information about learners' learning behaviors in online courses, and teachers can analyze this information to obtain information about learners' performance and problems they may encounter in learning the course [13, 17]. However, most teachers often do not know where to start and do not have a clear idea of what they need to analyze, so we hope to develop a methodology that will guide teachers in the development of a complete learning analysis process, and that our analysis will cover a variety of levels of analysis, helping teachers to analyze from a variety of perspectives.

It is also important for teachers to have an intuitive and clear dashboard of analysis results that will help them to quickly take control of the information in the course to identify problems and make decisions. Therefore, we will need an analysis environment that not only provides teachers with visual analysis results but also helps to save time and cost in handling large and messy logs. Compared to other learning analytics research which focus on how to apply data mining or machine learning techniques [9, 12], our approach proposes a goal-driven methodology and a working environment. The objectives of this research are:

- Develop a methodology to guide the learning process of analysis where the analysis would cover various levels of analysis.
- Provide an environment where data can be processed, analyzed, and results displayed.

We chose to use the GQM approach to guide our learning analysis process through its top-down traceability relationships, Banerjee analysis as the analysis model to provide us with various levels of analysis, and ELK Stack as the analysis environment to help us process the data and perform analysis. Combining the above three elements, we summarize a set of thinking process, analysis approach and working environment.

The rest of this paper is summarized as follows: In section 2, we introduce the relevant research and background techniques on GQM, Banerjee analysis and ELK Stack.

*Corresponding Author: Nien-Lin Hsueh; E-mail: nlhsueh@fcu.edu.tw
DOI: 10.53106/160792642023032402016

Section 3 introduces our research methodology, including system architecture, data storage, and research objectives and problem design. Section 4 presents the results of our analysis from the actual case studies, which are visually presented on the web through Kibana. In section 5 we summarize the results of our entire study.

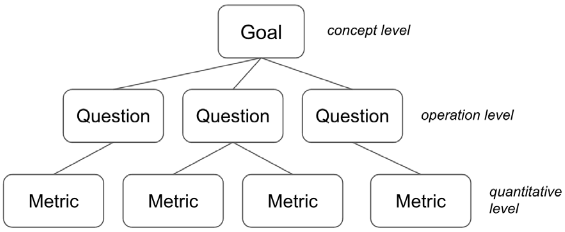

Figure 1. Goal Question Metric model

2 Background Technology and Related Work

2.1 Goal-Question-Metric Approach

Goal-Question-Metric (GQM) [5] is a goal-oriented approach to planning metrics that identifies one or more questions for a goal and determines which relevant metrics to use to answer those questions. This top-down traceability relationship helps us to focus our analysis on useful and valuable metrics. As shown in Figure 1, GQM defines a metric model at three levels:

- Conceptual level (Goal): the definition of a goal for a measurement object from various perspectives for various reasons, which can be a product, process, or resource.
- Operational level (Question): a set of questions used to describe how a particular goal is evaluated or achieved based on some characteristic model. The questions attempt to characterize the measurement object (product, process, resource) based on selected quality questions and determine its quality from a selected perspective.
- Quantitative level (Metric): a set of data associated with a question to answer it quantitatively. The response metric can rely solely on data descriptions of the object of observation such as the number of courses and number of videos, or it can be combined with a viewpoint to obtain an objective evaluation such as text readability and user satisfaction.

The GQM approach constructs objectives through four dimensions: purpose, issue, object, and viewpoint. We can obtain the purpose and the issue to be analyzed by analyzing information such as policies and plans of the company or organization. The object refers to the information to be analyzed is usually a certain process or product, and the viewpoint is the member who has a stake in the process or product, and the analysis from different viewpoints can obtain information that is valuable to each of them.

GQM is very commonly used for planning metrics, and many studies have used GQM to assess whether the metrics obtained can be used for objectives. Neubrand and Haendler use GQM to assess the maturity of DevOps in organizations and suggest that objectives can be subdivided into sub-objectives so that questions with metrics can be derived to assess specific objectives [10]. Tahir and Arif used GQM to create and evaluate a list of questions applicable for children's mobile education [14]. Abduldaem and Gravell worked on the use of tools such as Business Intelligence (BI) and dashboards to improve the accuracy and efficiency of available data by combining GQM and Balanced Scorecard (BSC) to generate BI-compliant dashboards to improve performance and support decision making [1]. Using the GQM approach helps us list the questions that should be analyzed and gives us the metrics to use to answer them, helping us decide which metrics to present in the Kibana dashboard.

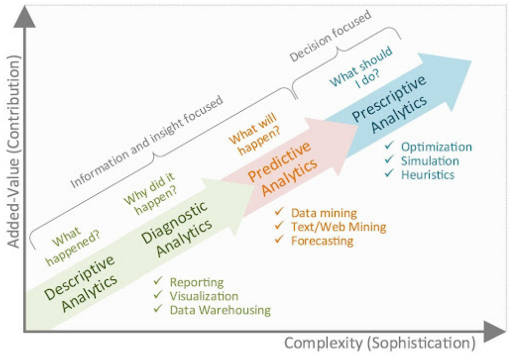

Figure 2. Banerjee analysis model [6]

2.2 Banerjee Analysis Model

Data Analysis is the process of using statistical methods and techniques to analyze data to discover causal relationships, internal linkages, and business rules. Holsapple, Lee-Post, and Pakath classify the study of analytics into domain, technique, and direction, with the domain dimension referring to the subject area applied, the technique dimension emphasizing the techniques used to analyze the data, and the direction dimension referring to the direction of thought [7]. The direction dimension is the most common taxonomy of analytical techniques, and Delen and Zolbanin divided it into three dimensions: descriptive, predictive, and prescriptive analysis [6]. Banerjee, Bandyopadhyay, and Acharya added one more diagnostic analysis after descriptive analysis and further divided the direction of analytical techniques into four dimensions [3]:

- *Descriptive analytic* is the most basic level of analysis, which is used to understand what happened in the past and to ask simple questions, emphasizing the analysis of "What happened?".
- *Diagnostic analytic* emphasizes the analysis of "Why did it happen?" and digs deeper into the descriptive data to understand the underlying reasons behind what happened.
- *Predictive analytics* emphasizes "What will happen?", learning from data through machine learning algorithms and building predictive models to discover and predict future trends.
- *Prescriptive analytic* emphasizes "How can we make it happen?", which uses the data presented

to plan specified measures or recommendations to achieve a favorable outcome. As shown in Figure 2, as the level of analysis gets higher, the value and difficulty of achieving it increase.

The Banerjee analytical model can guide the analyst to make appropriate decisions through different levels of analysis. Achenbach and Spinler [2] used the Banerjee analytical model to predict the arrival time of short-haul flights and optimize the cost index, Berk et al. [4] used the Banerjee analytical model to plan optimal human resource allocation decisions for professional service industries, Ito and Fujimaki [8] used the Banerjee analytical model to analyze and develop optimal pricing strategies for their products. They both successfully used the Banerjee analysis model to obtain the decision they should take.

2.3 ELK Stack

ELK Stack is a decentralized indexing and data processing platform. As shown in Figure 3, ELK Stack is mainly composed of four different components: Filebeat, Elasticsearch, Logstash, and Kibana. Each component has a specific purpose to help us manage big data and assist in analysis. The specific functions of each component are as follows:

- *Filebeat* is a lightweight delivery program for forwarding and centralizing log data, primarily monitoring specified log files or locations on the machine, collecting log events, and forwarding them to Logstash for parsing and indexing.
- *Elasticsearch* is a distributed search and analysis engine based on Apache Lucene that works on all types of data, including text, numbers, geospatial, structured, and unstructured data, and extracts the information we want to know by searching and computing on the data.
- *Logstash* is a tool for data extraction that collects data from multiple sources and supports processing different types of files, filtering and parsing the data, and sending it to the Elasticsearch repository.
- *Kibana* is a free and open front-end application that provides search and data visualization for data indexed in Elasticsearch, and serves as a user interface for monitoring, managing, and securing ELK Stack clusters.

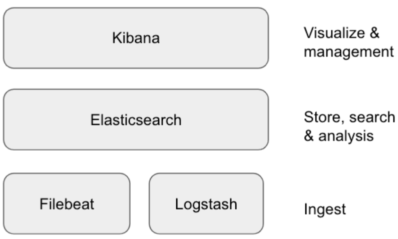

Figure 3. ELK Stack architecture

ELK Stack is widely used in enterprise and big data analysis research due to its millisecond search response, its ability to quickly retrieve information from massive log data, and its ability to provide data visualization dashboards. Tala ş, Pop, and Neagu used ELK Stack to obtain and analyze valuable traffic accident and earthquake data information from big data of cities for making city management decisions [13]. Persada, on the other hand, collected people's posts related to online learning on Twitter and used a plain Bayesian algorithm to perform sentiment analysis to understand what kind of words and phrases have positive and negative sentiments [11]. Yang et al. used road usage data provided by the Electronic Toll Collection (ETC) system to present real-time highway traffic conditions and to deeply analyze local road usage data. They successfully solved the problem of collecting and processing big data through the framework of ELK Stack and used the extracted data for further research and analysis, presenting the analysis results in a visual way for the researchers to view [16].

2.4 Related Work of Educational Data Analysis

Romero and Ventura identified two kinds of educational data analysis: Learning Analytics (LA) and Educational Data Mining (EDM) [12]. The main goal of LA is to improve learning processes while the main goal of EDM is to analyze data from educational systems. Both communities share a common interest in data-intensive approaches to educational research and share the goal of enhancing educational practice. In the survey, the authors have reported some LA/EDA systems for analyzing learning data. For example, DataShop provides a central repository with security and a set of reporting tools. The tool GISMO in sourceforge.net provides a graphical interface and useful visualization of students' activities. MDM tool provides a framework for applying data mining techniques in the Moodle system. Although these tools provide powerful features, they lack a customized, top-down mechanism to guide the analysis compared to our goal-driven approach. Abduldaem and Gravell [1] also pointed out the benefits of a goal-driven approach in education data analysis, however, they only identified the principles that were not implemented as a system.

Many data mining methods are applied in the online learning area to explore how students learn and improve the teaching mechanism [9, 12]. For example, casual mining for finding what features of students' behavior cause learning and drop out; clustering for grouping similar materials or students based on their interaction patterns, prediction for predicting student performance, and statistics for analyzing, interpreting, and drawing conclusions from educational data. Our proposed method in this research does not focus on the mining method, but the mechanism to guide the analysis goal, and a real system to conduct the and visualize the data. Compared to the ANALYSE system [11] which also analyzes edX log data, our system is built on the top of ELK and easier to scare up and maintain.

3 Research Method

3.1 Research Framework

Our overall research framework is sequentially divided into the following steps:

1. identify the goals to be achieved or satisfied for a specific role in the course;

2. explore the question by means of the Berjanee model;
3. indicate the metrics to answer the questions and the data collection ways;
4. find out the best way to visually represent the metrics and indicators.

We first use the GQM method to construct the main analysis objectives through the four elements of purpose, issue, object, and viewpoint. Among them, the viewpoint is especially important. Analysis objectives formulated from the perspective of stakeholders can effectively bring benefits to them, such as teachers and students, who seek different benefits and therefore have different analysis objectives. For example, from the teachers' perspective, the goal is to understand the watching rate of a video, but from the students' view, they want to know if they study harder than other students. The following is an example of the goal "to understand the learning outcomes of online learning behaviors from the teacher's viewpoint".

- Purpose: understand
- Issue: learning outcomes
- Object: online learning behaviors
- Viewpoint: teacher

After developing the objectives, we applied the Banerjee analysis model to propose the questions related to analysis. Goals can be decomposed into sub-goals if the questions can't be proposed easily. In the example above, some questions are proposed:

- How are students' final grades? (DESC)
- How do students learn online? (DESC)
- Why so many students can't pass the course (DIA)
- How to early predict students' outcome from the early behavior? (PRED)

Here DESC, DIA, PRED and PRSC refer to descriptive, diagnostic, predictive and prescriptive analysis in the Banerjee model. To answer the questions, some indicators or metrics may be proposed. For example:

- How are students' final grades? (DESC); the related metrics are:
  - fail rate of the course;
  - the distribution of students in different grade periods.

Another example is that teachers may want to reduce the dropout rate of the course. So, the goal "Reduce the dropout rate of the course" is identified. Then the following questions are proposed: "What is the current drop rate in the pass course? (DESC)", "Why do students drop out of the course? (DIA)", "What comes if we plan to decrease the dropout rate by reducing the difficulties of the homework? (PRSC)". To answer these questions, some metrics like "dropout rate" or "correlation between drop and learning behaviors" may be identified.

After obtaining the metrics, we will explore how to calculate the metrics we need and understand what data we need to use behind them so that we can have a basic understanding of the data we need to collect next, and then we can continue to explore how to obtain the data we need to collect. Take the OpenEdu e-learning platform as an example, most of the data we need to collect are stored in log files or databases. We need to think about how to extract the data needed for the metrics from the log files or databases and use the ELK Stack as the analysis environment to help us calculate the metrics. Finally, we decide on the visualization of the metrics to help the analysts understand our results easily and quickly through graphs.

3.2 System Architecture

In terms of system architecture, as shown in Figure 4, our system consists of four main components, namely, OpenEdu MOOC system, ELK Stack system, external learning systems, and external analysis modules. The lines with arrows indicate the data flow between modules. OpenEdu generates a large amount of log data every day, which represents students' behaviors of learning. We collect the logs through the Filebeat in the ELK Stack system and transfer the data to Logstash for parsing and filtering, and finally store the data into the Elasticsearch repository. Logstash extracts heterogeneous data from external systems like Online Judge and Star Trek. In addition to the log data, Elaticsearch can integrate, count, and compute these data, and present the analysis results visually in the dashboard through Kibana.

We need complex computations or special analysis models, which is beyond the capability of Elasticsearch to handle, so we have added an advanced analysis module in an external environment to help perform analysis.

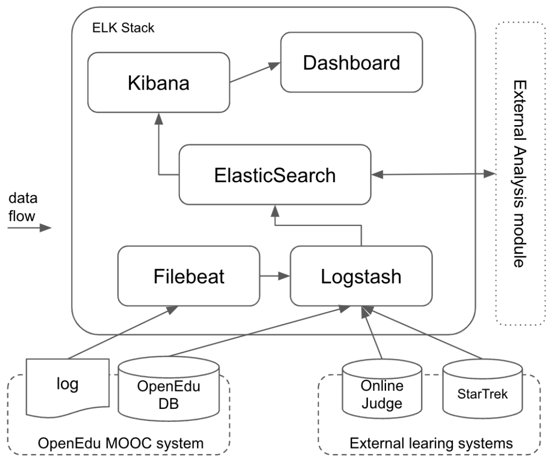

Figure 4. System architecture

ELK Stack Environment Building To facilitate deployment on different machines, we use docker-compose to build our ELK Stack system. First, we need to adjust the *vm.max* map count parameter of the Linux host to 262,144 to provide enough virtual memory for the ELK Stack system. Four systems including Filebeat, Logstash, Elasticsearch, and Kibana are built. The Elasticsearch cluster contains one main node and two data nodes, and the JVM heap of the main node is set to 1G, while the JVM heap of the data node is set to 2G, which can be adjusted according to the machine's memory. This can be adjusted according to the size of the machine's internal memory, but it should be noted that if too much JVM heap is allocated to your node, it will lead to long garbage collection periods. Next, adjust the output of *filebeat.yml* to Elasticsearch and *logstash.yml* batch transfer size and delay time settings respectively to ensure stable data transfer.

Data Extraction by Filebeat and Logstash Filebeat will transfer the collected log files to Logstash for parsing, and Logstash can write YAML files to regulate the rules of parsing and extract the data we want to analyze from the chaotic log data. After parsing, we save the data to the Elasticsearch repository for users to analyze. Table 1 details the event logs we extracted from the log data and what they represent. These events contain information about videos, quizzes, and forums that we can analyze to understand what happened to students in the online course.

Table 1. Event logs and their trigger condition in Open edX (partial listed)

Events	Description
Play video	Triggered when the user presses the video play button.
Pause video	Triggered when the user presses the pause button.
Stop video	Triggered when the video reaches the end of the video.
Seek video	Triggered when the user pulls the video progress bar.
Speed change	Triggered when the video playback speed on the user's player changes.
Problem check	Triggered when the user checks the problem.
Problem check fail	Triggered when the user fails to check the problem.

Computation by Elasticsearch Elasticsearch can search and compute the data to get the information we want to know. Elasticsearch can be structured as a cluster with multiple nodes, and there are many different roles for nodes. However, a cluster must have two types of nodes: a master node and a data node. The main node is used to manage cluster metadata information, cluster node information, and cluster index metadata information, while the data node is responsible for saving data and performing data-related operations, such as adding, deleting, searching, and aggregating data. Each node in the cluster can carry one or more slices, and each node has its master slice as well as the replica slices of other nodes so that if a node fails, other nodes can still provide the data owned by the failed node, which provides high availability (HA) for the system. In this study, we use a basic three-node cluster to build the Elasticsearch system, which contains one main node and two data nodes to avoid the failure of a node that causes the whole system to crash and lose the stored data.

In Elasticsearch, data is stored in JSON format as indices according to the type of data. Kibana provides an Index Management to help us monitor and manage indexes in Elasticsearch. We can observe the number of pieces, the number of documents contained, and the overall size of different indexes, and we can also merge and delete indexes.

To handle a large amount of data from various sources and to perform more advanced analysis, we divided the indexes in Elasticsearch into three layers in the data warehouse architecture to handle different levels of data: Operational Data Store (ODS), Data Warehouse Detail (DWD), Application Data Store (ADS). ODS layer is mainly used for storing unorganized log data and related data synchronized from the database; DWD layer is used for filtering out useful data after data cleaning and normalization and storing them separately by type; ADS layer is used for integrating or further analyzing data from the data storage layer to obtain data of a certain topic bias or a certain storage structure for visual graphical presentation in the Kibana dashboard.

Display by Kibana and Dashboard Kibana mainly uses the Kibana Query Language (KQL) syntax to query Elasticsearch indexed data. We can query specific fields for keywords to get the corresponding data and visualize the data through the visual analysis templates provided by the Kibana Dashboard. Kibana provides us with a convenient UI interface to help us quickly manipulate Elasticsearch to perform statistical and aggregation operations on data, saving users the time and cost of writing complex Elasticsearch commands and making it easier to obtain the metrics that users want to understand from the data and present them visually in the Kibana Dashboard. As shown in Figure 5, we can freely choose from a variety of presentations, such as pointer, pie chart, histogram, line chart, radar chart, etc., to help users quickly grasp the information in the learning platform from the dashboard and identify problems from it.

Figure 5. Kibana visualization dashboard

External analysis modules Sometimes the analysis we want to perform requires complex computations or special analysis models to complete, which is beyond the original ability of Elasticsearch to handle. Therefore, we have added an advanced analysis module in addition to the ELK Stack to solve this problem, where you can freely choose the analysis method you want to use and the model to train the prediction model, providing the system with more diverse and rich analysis possibilities. At present, our advanced analysis module provides analysis methods including Pearson correlation analysis, ANOVA analysis, and some machine learning training models such as Random Forest, SVM, and so on. The data is computed in Python and the results are transferred to the Elasticsearch repository and presented visually through Kibana.

4 Case Study

We analyzed data from a Python programming course offered by the College of Information and Electrical Engineering at Feng Chia University in Taiwan, 2021. The goal of the course was to teach students basic Python programming, including computation, logic, collection objects, functions, and data analysis methods. The course was conducted in a blended format, which comprised 2 hours of online study anytime and anywhere and 1 hour of in-class discussion each week of the semester. To make online learning more interactive, we designed several types of learning materials, such as quiz-in-video activities that enabled the students to quickly refresh their learning, online programming system (called Online Judge; OJ) for code exercising, slide flashcards for refreshing the knowledge, and the Game-based assessment (called StarTrek) for fun and test. The course won an Outstanding MOOCs Award from Taiwan's Ministry of Education for its well-designed curriculum and application of technology.

Initially 334 students were registered to this course, most (82%) of the students were Information Engineering and Computer Science majors; the others were majoring in Materials Science, Applied Mathematics, Automatic Control Engineering, Chemical Engineering, Business Administration, Aerospace and Systems Engineering, Civil Engineering, Statistics, and Electrical Engineering. We can collect learners' performance in these learning modules to assess learners' learning effectiveness. Here we identify 3 goals from teacher's viewpoint in the course:

- G1: Understand students' learning status.
- G2: Take early actions to prevent dropout rate.
- G3: Improve the course.

4.1 Data Collection

We collected the learning history data of students in these three systems and used ELK Stack 7.10.1 to build a data analysis platform, using Python and data science as an introduction to analyzing the study.

OpenEdu logs data The OpenEdu learning platform generates many logs every day, and we collected a total of 32 GB of logs for 8 months from September 2020 to May 2021 for analysis.

Heterogeneous data import and integration To help students learn better in online courses, we have developed the Star Trek gamification learning module and the Online Judge program learning module. To integrate these data, we also created two analysis pipelines, *StarTrek.conf* and *OJ.conf*, to help us parse the heterogeneous data from the Star Trek CSV file and the OJ database and manage the maximum number of threads used by different pipelines to parse the data through *pipeline.yml*. This prevents Logstash from overloading the machine during data transfer and parsing.

With Logstash we can import the Star Trek CSV file and the Online Judge database into ELK Stack, and it is important to note that the connection to the database requires a separate download of the JDBC Driver for the current ELK Stack version to help Logstash be able to connect. After the data transfer, we can integrate the indexed data through the python program Elasticsearch. We save the data as a new Application Data Store index for visual analysis by Kibana. Currently, the advanced analysis module includes Pearson correlation analysis, ANOVA variance analysis, and Random Forest machine learning methods, which can help us perform more diverse analyses, and more can be added in the future.

In the following we will explain the goals and their related questions and metrics.

G1: Understand students' learning status

To achieve the goal, we propose some questions and explore some metrics and figures to answer the questions.

What is the status of continuous learning? In a MOOCs course, usually students' participation decreases over time. We have the same problem in the course.To answer this question, we identify the metrics of video watch, OJ exercising and game playing. The video watch decreases slightly to 72.1% at the end of the course, while OJ and game are 38.6% and 61.5% respectively. This shows that watching videos is easier to do and students want to learn something from the course. Playing the game to finish the test is a little hard but they still like to do it. OJ exercising is most difficult, and many students stop doing that anymore.

Are the OJ questions too difficult for students? To answer this question, we check the submission to compare the number of passes and fail. Note that a student can submit many times before and even they pass the assignment. In Figure 6 we can see the failure rate in Unit 2 (control logic) is suddenly increasing to over 4,000 times in total and 19 times on each assignment. It means that the concept and practice of logic control is difficult for new learners.

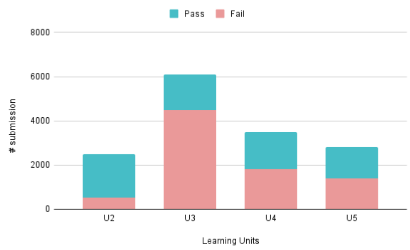

Figure 6. Number of submissions in each unit

What kind of errors are most likely to occur In Figure 7, we can observe that the most common errors that occurred during the OJ program practice were compilation errors and answer errors, accounting for 32.66% and 31.78% respectively, followed by execution errors of 21.17% and partial correctness of 13.33%. This result shows that the syntax is the most challenging for the new program learners. The compilation errors may come from those students not familiar with the Python syntax and how to complete a program.

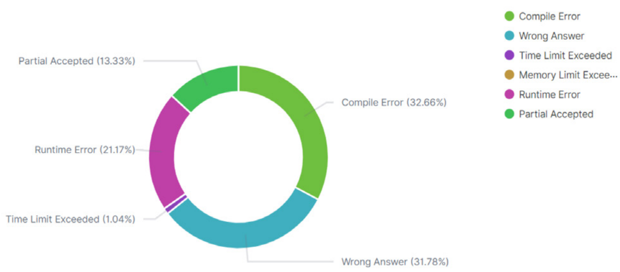

Figure 7. Error types

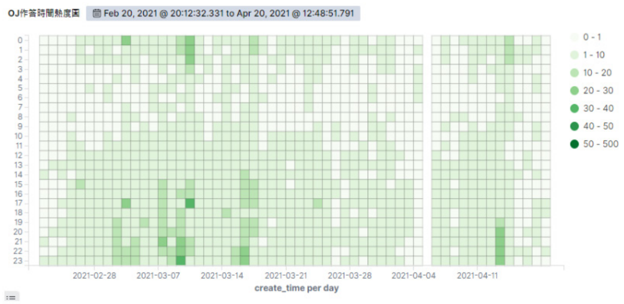

Figure 8. Time heatmap of students' OJ practice

When did students practice OJ programs? One of the benefits of online courses is that students can learn anytime and anywhere. Understanding the heatmap of practicing code can strengthen our belief and arrange the teaching assistance. The metric "Number of students studying at different time periods from Monday to Sunday" is proposed. Figure 8 shows that most students practice coding during the daytime, but the mid-night is more popular for most students.

G2: Take early action to prevent dropout rate

To achieve the goal, we identified the questions and metrics.

What is the correlation between different learning behaviors and grades? We use Pearson correlation analysis in the advanced analysis module to help us evaluate the correlation between different learning behaviors and grades, the data is first calculated in the advanced analysis module and then transferred back to Elasticsearch to be re-saved into a new data index. In Figure 9, we can see that the correlation coefficients between video viewing, question answering, and Lab participation and total viewing time and grades are all greater than 0.7, which means that these learning behaviors have a high positive correlation with grades, and the coefficient of correlation between the number of forum discussions and grades is close to 0, which means that the relationships between these behaviors and grades are weak.

Figure 9. The correlation heatmap between different learning behaviors and grades

How to predict whether a student can pass the course Predicting whether a student can pass helps teachers understand which students are likely to have learning difficulties, and then encourage or guide students for further learning. We use Random Forest to train the predictive model, using highly correlated learning behaviors from Pearson's correlation analysis as the training feature, including video watching, question answering and OJ exercising. 70% of the data were used as training data, and the other 30% were used as test data to evaluate the accuracy of the predictive model, from which a reliable predictive model was obtained to determine whether students would pass the course. To answer the question, we should collect the metrics from the students' behavior on the system.

From the performance analysis report of the random forest in Figure 10, we can see that the accuracy of our model in predicting failing students is 0.97, which is higher than the accuracy of predicting passing students of 0.94. This is because 72% of our training data belong to failing students, so it has higher accuracy in predicting failing students.

	precision	recall	f1-score	support
0.0	0.97	0.97	0.97	183
1.0	0.94	0.94	0.94	77
accuracy			0.96	260
macro avg	0.95	0.95	0.95	260
weighted avg	0.96	0.96	0.96	260

Figure 10. Random Forest evaluation

G3: Improve the course

In addition to the logs in the system, we also conduct the survey to understand the problems of the course and know how to improve the course in the next time.

How satisfied are students with different course materials? We collected feedback from 129 students to analyze their satisfaction with the different learning components of the course. From Figure 11 we can see that most modules such as videos, lecture notes, quizzes, flashcards, and online demos are preferred by students. However only 55.81% students liked the Quiz in Video component. After collecting students' suggestions, we understand that the quizzes in the video may interrupt their learning (watching video) and feel uncomfortable.

The acceptance of Online Judge is also relatively low. After deeper analysis on students exercising log, we found that students in "Unit 2: logic" encounter many problems in finishing the assignments. There are many submissions but returns fail (74% fail). The OJ system only notified them that they did not pass the exercise, but did not tell them how to solve the problem. Therefore, students dislike the learning modules.

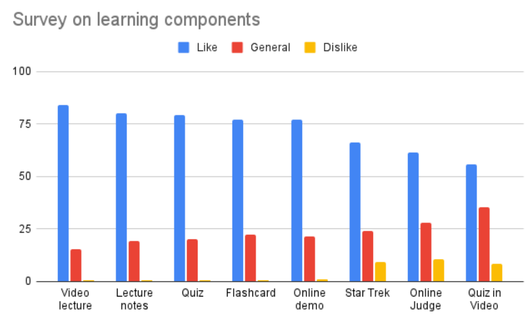

Figure 11. Survey on learning components

What are the main reasons for students to dropout of the course? From the analysis, we see that the main reason for students to stop taking the course is that they are too busy to study, accounting for 84.21% of all reasons, followed by the course being too difficult, accounting for about 10%. This means that the difficulty of the course is not a major problem. The biggest problem is that they do not have enough time or motivation to spend on the course. The low motivation may be due to the course being free and students can drop out from the course easily.

Do students recommend the course to other friends? From the survey, the student recommendation rate is 92.24%, meaning that most students think the content of this course is helpful to them and they can learn valuable knowledge from it, so most of them are willing to recommend this course to other students.

5 Conclusion

In this research work we have successfully built a learning analytics methodology and system with the following features: (1) Starting from a top-down exploration that guides the analysis process through a GQM approach, our approach highlights the advantages of a goal-driven and user-oriented approach compared to other systems that focus on data mining techniques and offer built-in analysis modules [9, 12]. (2) Instead of developing the system from scratch, our system is built on the top of ELK, which provides a resilient environment for us to collect, parse, and filter the huge amount of log data from the online learning platform, saving a lot of time and cost in processing the data. Moreover, compared to other edX analysis systems like ANALYSE [11] or learning analytics systems explored in [12], our approach is easier to integrate with other heterogeneous data from different sources through well-known ELK architecture.

In the future, we plan to integrate more tools to help students learn in the field of software engineering education. Students can learn related practices such as software testing or version control in online courses and environments, and improve their academic performance based on feedback from our learning analytics system.

Acknowledgment

This research was supported by the National Science Council, Taiwan R.O.C., under grants MOST110-2221-E-035-031-MY2.

References

[1] A. Abduldaem, A. Gravell, Principles for the design and development of dashboards: literature review, *Proceedings of INTCESS 2019- 6th International Conference on Education and Social Sciences*, Dubai, UAE, 2019, pp. 1307-1316.

[2] A. Achenbach, S. Spinler, Prescriptive analytics in airline operations: Arrival time prediction and cost index optimization for short-haul flights, *Operations Research Perspectives*, Vol. 5, pp. 265-279, 2018.

[3] A. Banerjee, T. Bandyopadhyay, P. Acharya, Data analytics: Hyped up aspirations or true potential? *Vikalpa*, Vol. 38, No. 4, pp. 1-12, October-December, 2013.

[4] L. Berk, D. Bertsimas, A. M. Weinstein, J. Yan, Prescriptive analytics for human resource planning in the professional services industry, *European Journal of Operational Research*, Vol. 272, No. 2, pp. 636-641, January, 2019.

[5] V. R. Basili, G. Caldiera, H. D. Rombach, The goal question metric approach, in: J. J. Marciniak (Eds.), *Encyclopedia of software engineering*, Wiley-Blackwell, 1994, pp. 528-532.

[6] D. Delen, H. M. Zolbanin, The analytics paradigm in business research, *Journal of Business Research*, Vol. 90, pp. 186-195, September, 2018.

[7] C. Holsapple, A. Lee-Post, R. Pakath, A unified foundation for business analytics, *Decision support systems*, Vol. 64, pp. 130-141, August, 2014.

[8] S. Ito, R. Fujimaki, Optimization beyond prediction: Prescriptive price optimization, *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, Halifax, NS, Canada, 2017, pp. 1833-1841.

[9] P. Leitner, M. Khalil, M. Ebner, Learning analytics in higher education—a literature review, in: A. Pena-Ayala (Eds.), *Learning analytics: Fundaments, applications, and trends*, Springer, Cham, 2017, pp. 1-23.

[10] T. Neubrand, T. Haendler, Development of a GQM-based technique for assessing devops maturity, *Proceedings of the 12th International Conference on Knowledge Management and Information Systems-KMIS*, Online, 2020, pp. 117-129.

[11] S. Persada, A. Oktavianto, B. Miraja, R. Nadlifatin, P. Belgiawan, A. P. Redi, Public perceptions of online learning in developing countries: a study using the ELK stack for sentiment analysis on twitter. *International Journal of Emerging Technologies in Learning (iJET)*, Vol. 15, No. 9, pp. 94-109, May, 2020.

[12] C. Romero, S. Ventura, Educational data mining and learning analytics: An updated survey, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Vol. 10, No. 3, Article No. e1355, May/June, 2020.

[13] J. A. Ruipérez-Valiente, P. J. Muñoz-Merino, J. A. Gascón-Pinedo, C. D. Kloos, Scaling to massiveness with analyse: A learning analytics tool for open edx, *IEEE Transactions on Human-Machine Systems*, Vol. 47, No. 6, pp. 909-914, December, 2017.

[14] R. Tahir, F. Arif, A measurement model based on usability metrics for mobile learning user interface for children, *The International Journal of E-Learning and Educational Technologies in the Digital Media*, Vol. 1, No. 1, pp. 16-31, January, 2015.

[15] A. Tala, F. Pop, G. Neagu, Elastic stack in action for smart cities: Making sense of big data, *2017 13th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, Cluj-Napoca, Romania, 2017, pp. 469-476.

[16] C. K. Tsung, C. T. Yang, S. W. Yang, Visualizing potential transportation demand from etc log analysis using ELK stack, *IEEE Internet of Things Journal*, Vol. 7, No. 7, pp. 6623- 6633, July, 2020.

[17] G. Veletsianos, J. Reich, L. A. Pasquini, The life between big data log events: Learners' strategies to overcome challenges in MOOCs, *AERA Open*, Vol. 2, No. 3, Article No. 2332858416657002, July-September, 2016.

Biographies

Nien-Lin Hsueh received the M.S. and Ph.D. degrees in computer science from the National Central University, Taiwan, in 1995 and 1999, respectively. He is currently a Professor with the Department of Information Engineering and Computer Science, Feng Chia University, Taiwan. Prof. Hsueh has served as the President of the Taiwan Software Engineering Association since 2017, dedicated to promoting software engineering education and technology in Taiwan. His research interests include software engineering, software testing, education technology, and learning analytics.

Jun-Jie Wang received the B.S. and M.S. degree in Computer Science from the Department of Information Engineering and Computer Science, Feng Chia University, Taiwan, in 2020 and 2022, respectively. His research interests include software engineering, education technology, learning analytics, and distance learning.

Daramsene Bilegjargal received the B.S. degree in broadcasting engineering from the School of Broadcasting and Media Arts, Mongolian National University of Arts and Culture, in 2012, and the M.S. degree from the Mongolian University of Science and Technology, Mongolia, in 2014. She is currently pursuing the Ph.D. degree with the Department of Information Engineering and Computer Science, Feng Chia University, Taiwan. Her research interests include education technology, learning analytics, distance learning, and software engineering.