Implementing Temporally Coherent Clustering on Student Activity for Probabilistic Knowledge State Determination & Optimizing the Learning Pathway through Markov Chain Analysis

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
August 2019

Implementing Temporally Coherent Clustering on Student Activity for Probabilistic Knowledge State Determination & Optimizing the Learning Pathway through Markov Chain Analysis

by

Ahmed Saquif Alam Anough
19341020
Md. Tahmid Hossain
14101148
Kazi Ruzlan Karim
18241050
Umar Faruk
15141003

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
August 2019

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work submitted as a portion of the coursework for completing our/my Bachelor's Degree at BRAC University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Ahmed Saquif Alam Anough
19341020

Md. Tahmid Hossain
14101148

Umar Faruk
15141003

Kazi Ruzlan Karim
18241050

Thesis Advisor's Approval

The thesis titled "Implementing Temporally Coherent Clustering on Student Activity to Predict Exam Performance & Optimizing the Learning Pathway through Markov Chain Analysis", submitted by

1. Ahmed Saquif Alam Anough (19341020)
2. Md.Tahmid Hossain (14101148)
3. Kazi Ruzlan Karim (18241050)
4. Umar Faruk (15141003)

of Summer, 2019 - has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on August, 2019.

Supervisor:

Hossain Arif
Assistant Professor
Department of Computer Science Engineering
BRAC University

Abstract

Recognizing a growing need to accommodate students of varied backgrounds and account for individual differences in learning curves, this paper reflects on our work to implement a Temporally Coherent Clustering approach in order to detect the most optimized pathway for teaching subjects through an MCQ based learning platform. Standard approaches towards extraction of student activity data typically detect similar behavior patterns and use simple statistical analysis in order to make predictions regarding their result. In reality, this causes high noise in the data that is temporally inconsistent and largely inaccurate. We proposed to work with an evolutionary clustering pipeline that can be applied to learning data that we have collected through our Intelligent Teaching System - and aimed at improving cluster stability over a large data set of student behavior. Initially, we have collected and worked on BCS Examination related data, where our results show improved cluster performance of both students and study material, and achieves stability on organic user data in order to be able to detect behavioral patterns and properties of learning environments. As an end result of this whole research, we have incorporated our work into our ITS, which proactively determines student's knowledge level, and automatically determines the best pathway in order to improve their performance. Overall, it deliberately influences a students capacity improvement in order to passively enable them to answer harder questions by creating an optimized pathway that recognizes the need for individualized learning curves. Overall, we managed to get an accuracy ratio of around 84%, with a silhouette score of 0.53 against an optimized k value of 5 within our clustering algorithm using k-means.

Keywords: Machine Learning; Temporally Coherent Clustering; Exam Performance Prediction; Result Prediction; Knowledge Level Determination;

Dedication

We dedicate this work to the millions of BCS aspirants across the country, who continuously struggle to balance their learning curves without any form of digitized tools to assist their endeavor. This is to make their lives a little easier.

Acknowledgement

Firstly, all praise to Allah (SWT) for blessing us with the health and knowledge that enabled us to cross this vast ocean of knowledge and perseverance.

Secondly, our gratitude to our Thesis Advisor Mr. Hossain Arif Sir for his kind support and advice regarding our work in-spite of his sickness throughout the past year.

Thirdly, a great amount of this work would not have been possible without the direct funding and mentorship support from the Center for Entrepreneurship at BRAC University. Starting from direct assistance in evaluation, and ending at DevOps support provided through Amazon AWS's Team - a lot of this Thesis depended on the work that CED helped spark off throughout these past years.

And finally, to our parents - without whom nothing would have been possible.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

BCS Bangladesh Civil Service

EMM Expectation Maximization Model

GDP Gross Domestic Product

HMM Hidden Markov Models

ITS Intelligent Teaching System

MCQ Multiple Choice Questions

OECD Organisation for Economic Co-operation and Development

Chapter 1

Introduction

The basic intention behind our research is to assist the development of an Intelligent Tutoring System (ITS) in order to proactively engage students within an action-oriented, hands-on learning process where their interaction with the gamified learning environment is used to foster maximal learning progress.

Using our dataset of student performance gathered from the Version 1 of our Application, we applied advanced and targeted data mining techniques to discover association, classification, clustering, and unexpected outlier detection rules.

Within each of these sepecified domains of study, we have presented our discoveries and understandings - and have used the knowledge to improve on our final solution that enables a pathway generation for improvement.

Throughout the next few subsections, we will explain why we decided to work with the mammoth task of collecting and working with our dataset and target scenario, and why this Temporally Consistent Algorithm is necessary to have a significant impact over the lives of a large subset of the population that can directly benefit from our work (excluding the factor that this Algorithm and the Principles behind our implementation are dynamic enough to be adapted to any scenario and any examination of the world, as long as a few parameters are kept constant).

Over the longer period, we hope to achieve a level of clustering accuracy that enables the development of an intuitive, non-manual, and non-perceptually-dependent Intelligent Teaching System that is capable of proactively determining a student's knowledge level by clustering them based on their initial performance, and then using statistical analysis in order to proactively suggest a pathway that can increase the efficacy of their studies.

Through a methodical approach derived from years of research into this project, our aim is to enable our Intelligent Teaching System to dynamically increase the vigor and intensity of the learning curve - thereby impacting the end-goal of passively maximizing the potential of each student through an avant-garde teaching process that is fully digitized, and yet personalized towards the specific needs and development curves of each student.

1.1 Background

Across Bangladesh, thousands of students attend the general public examinations in order to gain entry into the government and private jobs. These exams, by virtue of the amount of the examinees, are always taken as Multiple Choice Questions (MCQ) Exams by the different governing bodies that design and hold these examinations.

While the question pattern of these exams have undergone a certain level of standardization from the governing bodies of these exams, the study material of these exams have remained largely manual and based on past-examinee perceptions.

Currently, students are forced to manually overcome their challenges regarding their own learning curves, and do not have any easy access to Intelligent Learning Systems that assist them through their learning journey.

Rather, the students are limited to using large question banks that provide zero feedback in order to practise for their exams, which ends up being an extremely arduous task where each student is expected to "memorize" thousands of data points without any help from Intelligent Teaching Systems that could have made their work much easier.

1.2 Basics of the Bangladeshi Exam Scenario

The biggest Public Exam in Bangladesh is known as the Bangladesh Civil Service Exam, popularly known as the BCS Exam of the Government of Bangladesh.

This exam is widely attended by a large subset of the population that is greatly varied in terms of the demographics and the background of the attendees. Starting from Geology Graduates and ending at Medical Professionals, the BCS Preliminary Exam is typically attended by 400,000+ examinees in each of its iterations.

This examination is conducted keeping in mind the wide range of attendees, and is a general test of their mental capacity and their cognitive abilities.

Although this is always conducted through a single exam paper that is not dynamic, the Public Services Commission (PSC) of Bangladesh is widely known to set an exam paper that is both varied and extremely critical - which sets a bar that allows them to filter down and differentiate between different students.

While the effectiveness and the potency of the BCS exam is widely debated due to its nature of requiring large amounts of memorization capability, it is still one of the most important exams that Bangladeshi Graduates attend every year, and remains socially as a differentiating factor for success parameters.

It is widely known that at times, Public University Students prepare for the BCS Exam throughout their University life, even at the expense of studies within their own Department courses - simply because one BCS Success could end up changing their lives forever.

In the context of this Thesis work, the BCS exam is thus an appropriate target group (TG) that serves as a high-volume and high-intensity exam that is attended by a wide demographic across the nation.

Additionally, given the fact that the BCS examination is conducted with a non-specific exam set (originally, the BCS examination is held with the subjects Bengali, English, General Knowledge, Bangladesh and International Affairs, General Science, Mathematical Reasoning and Analytical Ability, Geography, and Ethics and Good Governance), which means that this Thesis work is based on a wide range of subjects and demographics (our initial data-set is based off users from all 8 divisional cities of Bangladesh, with one single inclination towards Dhaka as a major city, which was also expected).

Overall, the BCS Examination provides a wide dataset of information that is absolutely relevant, widely representative, and not systematically biased, thus enabling an interpretation that has a low margin of error, alongside providing a wide scope of direct usage in the real-life scenario.

1.3 Understanding the Solution Scope

During the development and data collection phase of this project, which spanned over a large portion of time (this project was initially launched in Fall 2017), we have undertaken multiple questionnaire based research initiatives in order to understand the need from the user's end regarding such a solution.

In one of our initial user feedback generation drives conducted through our app (conducted over the first 241 users who signed up into our prototype front-end web application), the following result was obtained regarding the user's expectation of a perfect Intelligent Teaching System that would enable them to learn better:

Expectation from the Solution	**Agreement**
1. Needs Continuous Performance Evaluation	82.3%
2. Should Display comparisons of performance	23.7%
3. Should Suggest what I should study	96%
4. Should Predict my result	64.64%
5. Should Provide exam tips	24.64%
6. Needs to have a Conversational Interface	12.1%
7. Should work on my Phone	72.3%
8. Must be a Free Solution	97.4%

Table 1.1: User expectation from the application

From this, we came to a certain set of determinations regarding the expectations that users have from our application and the usability expectations that are prevalent within this community of users.

For instance, more than 82% of our initial users requested that we provide them with a continuous performance evaluation, while around 23.7% of users requested for comparison of performance (compared to all of the users, or in between friends). This meant that users are aware that in the job arena, they are competing against the whole set of examinees, and want a better understanding of their position within the whole system.

On the other hand, a surprisingly 96% of the test group of users responded that they want the Intelligent Teaching System itself to recommend what to study, instead of going for the manual approach of studying something and then deciding what to approach further with. At one point, this was an opinion expressed by almost all of the regular users of the app - who wanted the application itself to recommend the best pathway towards learning.

Especially, one approach that was pointed out multiple times by the top scorers (users on the third quartile, with exceptional accuracy rates that are on the top 25% of accuracy ratios) was that the current books that were used by the examinees had a solution where the questions would become slowly more difficult instead of being in a random order. This enabled the students to slowly learn the tricks of the chapter that they were attempting, and did not overwhelm them with information glut.

This is an idea that we invested our effort into heavily afterwards, because it was an opinion shared widely by the community, including [[31]], where it was found that a gradually increasing difficulty within an Intelligent Teaching system is an optimal solution that is widely considered as helpful for the end-users' adaptability over the long term.

As a whole, this thesis concentrates on the complete set of solutions that are required in order to create a Machine Learning Model that is capable of leveraging newly developed tools and domain knowledge in order to predict the student's current knowledge depth, and accordingly create a pathway that would enable them to slowly rise up the learning curve.

1.4 The Solution

In simple words, our solution revolves around a Mobile Application we are developing, which is a gamified MCQ Question platform designed for BCS and Job Examiners in Bangladesh who want to study for these competitive exams from the comfort of their home.

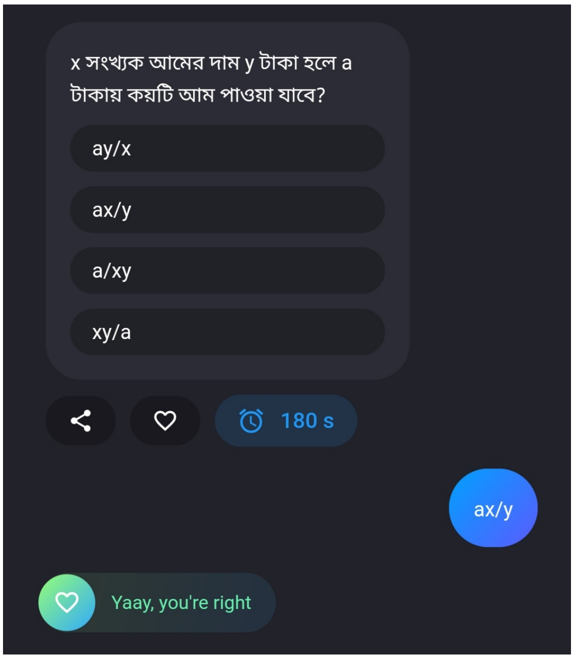
The way it works is that as a user, they can simply answer the questions that our chat bot ask - and based on the answers they give and the heuristics in our system, our Intelligent System will be able to predict their knowledge level - and accordingly serve them questions.

By creating an engagement platform that takes account of Csikszentmihalyi's Flow

Theory[[32]] and creates an incrementally increasing level of challenge, our aim is to objectively create clusters of different students into linearly dynamic and disentangled groups, so as to enable a personalized and adaptive machine-learning environment that can proactively judge a student's current knowledge level, predict their future score, and interact with them.

The idea is, instead of assuming the difficulty level of the questions being served or the student's knowledge level, we will rather create a dynamic platform that tries to understand what the student does not know, and serves to create a pathway of questions that will enable them to gain that knowledge.

The final result of our work will culminate into a large environment that is capable of collecting complex user behaviour such as cognition, affect, user trait, geographical similarities, perceptual cue effects, environmental variables, and even physical data in order to enable an interactive learning environment where every move is individualized by using machine learning strategies that determine the success rate of the next group of actions and thus guides the through their education journey.



x সংখ্যক আমের দাম y টাকা হলে a টাকায় কয়টি আম পাওয়া যাবে?

ay/x

ax/y

a/xy

xy/a

180 s

ax/y

Yaay, you're right

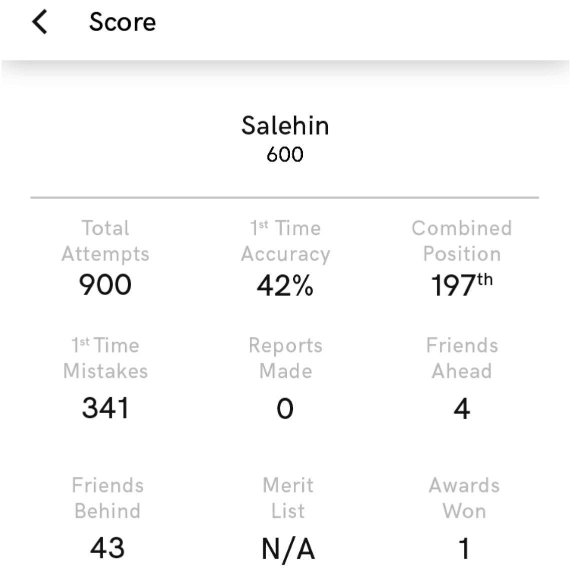
Figure 1.1: Example Question on the App

One extra attempt we made in order to make the system more capable, was to focus our research into developing a novel, independent, and unsupervised machine learning algorithm that can detect probability of developmental dyslexia though statistical analysis on our student performance over a long term based on cluster-shift rationality's.

But given the lack of adequate data sets of students who definitely suffer from a Development Calculi (by which we mean any kind of medical or professional opinion that guarantees or solidifies the proposition that we make regarding a student), we could not find any concrete deterministic model that would enable us to create a cluster that finds students suffering from such a medical condition. Over the longer time though, as our system becomes more intelligent and more ways open up to determine the learning pathways of students and recognizes outlier clusters of students who fail to improve on Mathematical subjects regardless of the growth pathway offered to them over a long term - we can concentrate on this proposition and its impact on Intelligent Teaching Systems.

Additionally, our system also provides a set of visual, emotional, and statistical tools in order to enable better insights into the current status of each student, alongside a comparison with respect to their immediate cluster performance.

For instance, instead of only depending on the score, we will also provide multiple other statistical data to the student. Such as:



Score

Salehin
600

Total Attempts 900	1st Time Accuracy 42%	Combined Position 197th
1st Time Mistakes 341	Reports Made 0	Friends Ahead 4
Friends Behind 43	Merit List N/A	Awards Won 1

Figure 1.2: Statistical Data shown to users upon continued usage

1.5 Why Temporal Clustering

One of the assumptions that any Machine Learning based clustering approach makes is to assume that similar behavioral patterns may be found within sequential data. Standard approaches simply identify clusters at each time separately, and show low performance from noise. This means that the data is temporally inconsistent.

In the situation of our application, it is important to recognize that our students can never be in a single cluster for a constant period of time - and rather, continuously change over a small period of time depending on the nature of the data that the student faces within the application.

It is even possible that the same student will jump across clusters within a time period as short as 1 hour (as observed in our analysis), which makes current methods of clustering incompatible and discrepant with the mutual goal of understanding student performance on a dynamic basis and taking steps based on that.

This is because if the clusters only make sense for a single period of time and must be re-done in order to understand the situation better at a later date, then a dynamic system cannot handle the processing load of such a scenario and yet continue to perform without interruptions of the service. This is one of the major reasons why studies related to Moving Networks [[33]] or Brain Function Analysis [[34]] make use of Spatio-Temporal Clustering in order to determine homogeneity and heterogeneity within data-points - since they share the same ideology of working with continuously moving data-points.

Therefore, we designed a Temporally Coherent Clustering model that accurately captures relevant cluster evolution effects and thus paves the way for a pipeline that may be used for capturing the dynamic patterns within a learning environment by fully exploiting the network characteristics in spatial, temporal and thematic domains of an Intelligent Teaching System.

1.6 Thesis Outline

Over the next few sections, we will firstly go through an overview of our Literature Review in the related fields - including prominent papers on Unsupervised Learning, Intelligent Teaching Systems, and Temporal Clustering Algorithms.

Afterwards, we will provide a brief review of the Algorithms, Datasets, and the tools that we have used throughout the Thesis period, including details on the challenges faced regarding deciding which algorithm worked best for us, the libraries that we used.

The Methodology section details the System Architecture of our ITS, and goes through the framework design of our application - which is closely followed by our Unique Interaction Design principles that enabled this whole project to come to life. Afterwards, we explained our work behind the manual clustering methods that we explored, including our work related to Knowledge State Determination and all other kinds of clustering approaches that enabled the Knowledge State Determination to work properly. Afterwards, we explained our work with the Prediction and Interpretations, and how we incorporated Temporal Information within our system.

This is then followed by our complete implementation and results section, where we detail the experiments and optimization that we have done throughout the process, including how we implemented the Temporally Coherent Clusters and what are the performance evaluations of this implementation.

The results and the Future Work section is a combination of our complete achievement throughout this whole journey, and describes where we are in the journey of developing a truly cognizant and dynamic algorithm designed for the students across various domains of study.

Chapter 2

Literature Review

2.1 Unsupervised Learning

Data mining is the practice of examining large pre-existing databases in order to generate new information. Data mining techniques [28] are used to explore the important pattern in data sets and also improve the way our algorithms can predict the results of future input points.

The division of unsupervised data into groups of similar objects is known as Clustering [29]. It models data by clusters. Ideally, an Unsupervised Clustering algorithm creates a k number of clusters, which must be set/determined manually. While it seems logical to create a large amount of clusters, creating fewer clusters means that finer details in the data are lost. Consequently, it also simplifies the clusters.

Data modeling puts clustering in a historical perspective [43] rooted in mathematics, statistics, and numerical analysis. From a machine learning perspective, clusters correspond to hidden patterns, the search for clusters is unsupervised learning [40], and the resulting system represents a data concept. In Unsupervised models, data only consists of input vectors, with no specific target output in mind. The algorithm figures itself out about what it should be looking for, in a sense it independently finds structure within the data set.

In this particular paper, we are focusing on Centroid-based clustering. In Centroid-based clustering [42], clusters are represented by central vectors, which may not necessarily be a member of the data set.When the number of clusters are fixed to an arbitrary value of k.

K-means clustering [30] is the most popular forms of clustering methods algorithm that continuously partitions the data set into K groups in the presence of only input vectors without referring to known, or labelled, outcomes.It is the most widely used for its simplicity and ease in training complex datasets.The algorithm aims to minimize the objective functions in this case the Square error function. .K-means partitions [41] n observation into K clusters in which each observations belongs to a the cluster with nearest mean.

Another kind of clustering which was introduced is Fuzzy C-means Algorithm [40].Fuzzy c-means clustering allows one data element to belong to two or more clusters.Given a finite set of data, X, the problem of clustering in X is to find several cluster centres that can probably characterize relevant classes of X. In classical cluster analysis, these clusters are required to form a partition of X such that the degree of association is strong for data within the block and weak for data in different blocks. However, this requirement is too strong in many practical application and it is thus desirable to replace it with a weaker requirement.

2.2 Temporal Clustering

Although K-means provides optimal clusters in sequential data it immediately falls short in data that changes over time.The problem of clustering sequences of unlabeled points sets taken from a common metric space is solved by Temporal Clustering [38].Temporal Scenarios arises in many application such as biological survey , contagious disease surveillance and in our case a student preparing for an important examination.Temporal Clustering can be achieved easily using three simple parameters: the number of clusters, the spatial cost and maximum cluster displacement between consecutive time steps.

In this paper, we propose using temporal clustering to learn about student activities in controlled environment which was previously used in an uncontrolled environment using evolutionary clustering pipeline [2]. They offered a solution in which the user can heavily skew the result of their ITS resulting in data with high noise .Although the noise can be normalized over multiple training session but this results in creating an inefficient approach that slows down the rate at which deduction can be made about the students participating in the program.On the other hand , this approach produces highly performent and stable clusters.

Another way of dealing with sequential data is using a Bayesian Knowledge Tracing model [5] on student data which stores what a single student has learned previously and then builds up on it.Although it produces better results than conventional methods it too comes with some drawbacks.The system assumes that for a single student ;prior learning is assumed to be common across different subjects but in reality it differs across different subjects.Furthermore, each student is exceptional than the other so the group for the data cannot be considered as a homogeneous groups.This model provides insight as to how we can use previous data to effect the how clusters are made.

On another paper [4] dealing with a large number of students nearly 3 million.The method of dividing student who used similar processes has proved to improve process mining results. This paper suggests using a sequence of clustering algorithm that groups student using different learning pathways.A graph based process discovery algorithm was used know a procedural steps that students took to learn a particular subject. It provided a graph which represented all the different ways a student take to study.

2.3 Intelligent Teaching System

Intelligent tutoring systems are clearly one of the successful enterprises in AI.There is a long list of these tutoring systems that have been tested on humans and have been proven to facilitate learning. For example, some ITS are well-tested tutors in algebra, geometry and computer language [35]. Various Computational modules are used in these ITSs of the world in AI. According to recent valuation, modules used from state of the art AI produces learning gains in people of approximately 0.3 to 1.0 standard deviation units compared to students who learn using traditional classroom methods [36].

In the next few decades ITS is expected to replace these outdated teaching methods and there are already signs of this changes in some parts of the world where students can easily earn a diploma or online credit by using ITS that are setup on the web.E-learning has become commonplace [45] in higher education and one such method of creating an ITS is using MCQ based questionnaires.When students attempts multiple-choice questions they generate invaluable information which can form the basis for understanding their learning behaviours.In this research, the information is collected and automatically analyzed to provide customized and diagnostic feedback to support student learning.

In a paper [3] that analyzed 1.08 million student data from Edulab discovered the different underlying student behaviors in the learning process.They analyzed the sequence of actions that each student took to complete a task.The student behaviors were modeled as Markov Chains whereas a student is modelled as a distribution of Markov Chains which are estimated using a modified K-means clustering algorithm.The model resulting from this experiment was illustrative and provided qualitative analysis.It was determined that around 125,000 student were very unproductive. However, the data also provided knowledge about how differently the ITS was used and helped them study alternative to common traditional practices.

The clustering of sequential facts may be a frequent strategy to detect similar behavior patterns and has been expeditiously utilized to a range of functions like analyzing comprehensions[8], online collaboration instrumentation [9], table-top environments [10], internet browsing [11], physics simulations [12] or homework assignments [13]. Moreover , a range of distinctive student behavior has been investigated . [14] identified students that impose challenges for the student models. Other work studied the relation between interaction patterns and also the overall performance of scholars [15],[16] and also the relation between student action sequences and their effective states [15].

Common strategies for the analysis of sequential data consists of sequence mining [17],[18], differential pattern mining [13] or Hidden Markov models(HMM) [19],[20]. Sequential sample mining strategies have been contextualized the usage of piece wise linear segmentation [21].

Other methods have employed the usage of semi-supervised graph clustering and two the usage of the predictions from a student model as additional constraints [14].

Clustering sequential facts employing similarity measures on state sequences was once used in [12],[22].

These state sequences can be aggregated into Markov Chains modeling the state transitions[23]. HMM have been also employed to extract steady groups from temporal data by means of joint optimization of the model parameters and cluster count[24].

All these previous work only focuses on clustering student data on a specific point in time. But a temporal analysis of the student data will help us identify how the interaction with the student and their pattern changes over time.

Collecting a temporal sequence of data from the students activity in a controlled ITS would help us identify how a student evolve with time over the given study material and help us to group similar students. Temporal effect of cluster evolution has been analyzed in [25] based on static clustering after a fixed amount of time stamp.

The problem with static clustering is that the data is susceptible to noise and results in inconsistent clusters. Evolutionary Clustering Methods address [26] this problem as the temporal smoothing increases the resulting cluster stability notably and allows for a better analysis of the clusters, i.e., the student properties and interaction patterns.

A new form of clustering approach called AFFECT [27] has been introduced that smooths the noise and cluster inconsistencies over time, and an evolutionary pipeline has been created using the AFFECT clustering method [1] which can be used as a black box for any ITS systems. It can improve the cluster stability over time and smooths out the noise in the student data.

Our approach includes using ideas from this [1] black box in order to create our own intelligent tutoring system that can serve students controlled questions based on difficulty set by the evolutionary clustering method.

2.4 Markov Chain

Being a Mathematical System/Model, a Markov Chain is simply a rule that allows a graceful transition from one state to another depending on given probabilistic rules [48] . The defining character of this chain methodology is that no matter how the process arrives at a certain state, the possible future states for this process is fixed and constant (with a probability that can be calculated) [49].

At the end of the day, it describes a stochastic model where the probability of each model depends on the state attained in the last event only (and not the ones before that).

Markov Chains are an extremely varied branch of mathematics that have been used for complicated implementations such as Google PageRank Analysis [50] to finding notes with high amounts of internal connectivity [51] in the Applied Mathematics field. This is a branch of non-linear science that has even been applied for the probabilistic description of the drift of marine debris (for the search of a plane crash scene) [52], proving that a Markov Chain analysis is a versatile field of analysis that may be implemented in wide fields of probabilistic approximations.

For the assessment of student performance, a Markov Chain is a widely used tool in the scientific arena, as explored in [53], [54], and [55].

When it comes to adaptive electronic assessment of students, the transition probability matrix was best described and used in [56], where the research used the exact same type of application scenario as our Application, and implemented a Markov Chain Analysis.

Additionally, much work has been done regarding using the Markov Chain analysis in the Higher Studies Domain, including [57], and [58], both of which implemented the concept for the case of Undergraduate and Higher Education domains (which is the target scenario of this thesis paper).

Chapter 3

Algorithms, Datasets and Tools

3.1 Algorithms Used

3.1.1 K-means

Usually, unsupervised algorithms are known to make inferences from the given dataset using only input vectors - which means that they do not refer to currently labeled outcomes (even if the labelling is domain knowledge and true by virtue of the dataset).

Now, since a cluster is a collection of data that ”looks” similar by virtue of mathematical similarities between them - it means that the objective of a simple clustering algorithm like K-means is to simply group similar data points into clusters through discovering underlying patterns that may not be easily visible to the eye.

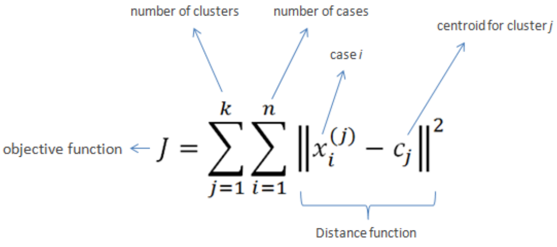


Figure 3.1: Objective Function of the k-means Clustering Approach

The k-means clustering algorithm does this by looking for a fixed number of dataset, which is represented by the k. The idea is that the user sets a number k, which is then understood to be the number of centroids (centers of each cluster) that the algorithm will try to find. Since this is not known as a-priory, it must be computed directly from the data and then interpreted for accuracy level optimization.

This is simply done by first taking a group of randomly selected centroids, which are then used as a beginning point for every cluster. By using repetitive iterations

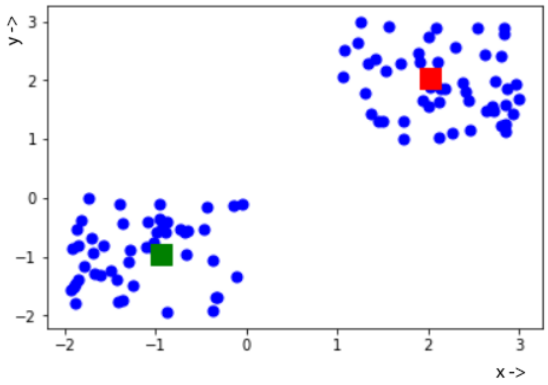


Figure 3.2: A typical k-means clustering algorithm result with the centroids displayed as squares

on these calculations, the k-means algorithm achieves an optimized position of the centroids with the end goal being either:

- Stable Centroids, where there has been no change in their position, which indicates that the clustering has been successful and that there is no need for further calculations.
- A certain number of iterations have been completed (usually done in order to limit the number of iterations that the system will try before considering the result as more or less stable.

However, the k-means approach is a generally inefficient method that only assigns objects to their closest cluster center according to the Euclidean Distance Function and then only calculates the centroid's position through finding the mean of every object in that cluster.

This is thus not an optimal solution since Euclidean distance can only be used for data-sets that are differentiated by having co-ordinates of where they are placed. This requires a manual interpretation that may or may not be applicable for all types of data, and is thus an extremely naive approach.

In our methodology section, we have explored the difficulties we have faced regarding Hamming Distance and Euclidean Distance, and which we have chosen to work with depending upon the scenario where they were implemented.

3.1.2 K-modes

One of the other options that we explored was the k-modes algorithm - which is generally differentiated by the fact that a k-modes algorithm uses modes instead of means to form clusters of the categorical data.

The k-modes algorithm provides an alternative to k-means when the data is categorical instead of numeric. Given the fact that k-means explicitly minimizes the inter-cluster variance (squared distance), it cannot be used for situations where the data is not numeric in its true sense.

A typical k-modes implementation first starts by randomly assigning a k number of data points as modes (this means that the system does not require a manual input of the number of centroids we need).

It then calculates the dissimilarity score between each of the remaining data points from the k number of chosen modes. This is then followed by associating the data points to the mode whose score is minimum, giving us a K-number of clusters.

$$d(X,Y) = \sum_{j=1}^{m} \delta(x_j, y_j)$$

where

$$\delta(x_j, y_j) = \begin{cases} 0, & x_j = y_j \\ 1, & x_j \neq y_j. \end{cases}$$

Figure 3.3: The Hamming Distance Function used by the k-modes Algorithm

Afterwards, the algorithm uses a ”Moving Mode Frequency Based Method” to update the modes (for each of the k clusters).

This whole process is then repeated until there is no reassignment of the clusters (or when the cost function of the iteration is minimized).

3.1.3 Customized Markov Chain Algorithm

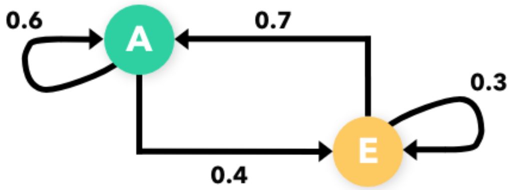


Figure 3.4: An example of a sequence of events that exhibit the Markov Property of reliance on the state that precedes it.

A Markov Chain is a mathematical object usually defined as a family of random variables (appears to vary in a random manner), and carries a property called the Markov Property, which says that its future/next action is not dependent on its past actions.

Named after Andrey Markov, it is a mathematical probability based system that allows ”hops” from one ”state” to another depending on a few criteria (a-priory).

Markov Chains are modeled by finite state machines, they arise broadly in statistical and information-theoretical scenarios - with implementations ranging in a multitude of various scenarios such as Game Theory, Queuing, Genetics, and Finance

Given that our research largely bases its roots in being able to shift a student from one state to another, this is actually an algorithmic problem best defined in the Markov Chain Model.

$$\mathcal{P}_{ss'} = \mathbb{P}\left[S_{t+1} = s' \mid S_t = s\right]$$

Figure 3.5: The Markov Probability Function, showing that the probability of an event happening in the future state is dependent on the current state

Our implementation of the Markov Chain has the defining characteristic that no matter how the student arrived at their present state (which is defined as their current cluster), their possible future state is fixed to a single cluster upgrade.

Or in other words, the probability that the student will transition to the next state or the state before is solely based on the current state and the performance achieved in its state (unlike the traditional model of time elapsed). For us, the state space is defined as the Track where their particular Markov chain is defined.

3.2 Datasets

3.2.1 Data Collection and Processing

Over four months of the initial Version 1 Release of the app (aimed at data collection only), our Android implementation collected a multitude of data-points directly from the user, including their answer times, accuracy ratios, location, contact list, access to other educational apps, etc. and use that data in our unsupervised learning algorithm.

From the start of the system-design, we decided to not only use Unsupervised Machine Learning in order to come to decisions - but rather, our business logic will largely be driven by the insight gained through Machine Learning - while still making use of our internal heuristics driven by logical assumptions, user-experience expectations and goals, and our goal-driven approach towards learning.

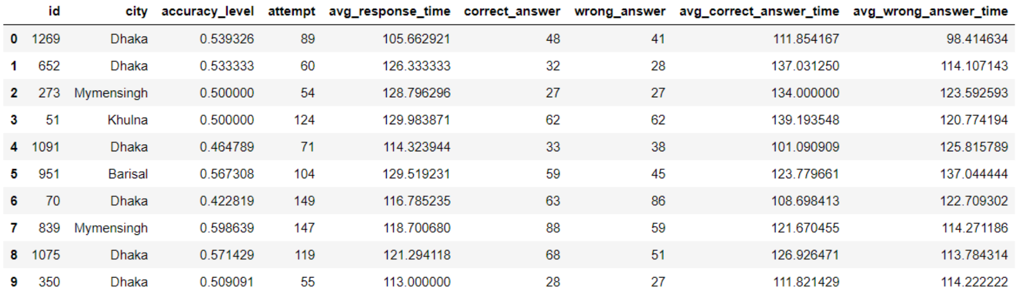
In order to achieve this target - over the past year, our team has manually cleaned a large portion of our dataset of previous-year's structured exam questions, amounting to over 4,000,000 characters - which will be used in our question serving logic throughout the ITS, was processed using our Clustering Algorithm and internal heuristics in order to positively determine their knowledge level.

In a very simple scenario - let us assume that 400 students have been served a question over the last one month. Our system will take into consideration as to how many of the students found this difficult based on multiple criteria.

Such as, if a large portion of the students answered the question wrongly on their first try, and correctly on their second try with a significant gap in between, then the system will assume that it is a case of a confusing question.

But, for instance, if the same question has a lack of parity in the answers - we will try to develop clusters of students for each question level - based on the amount of time they took per question, their personal accuracy ratios in that particular subject, their accuracy level in similar topics, their location, their relation to the students who failed to answer the questions correctly, etc.

With all that data, we classified whether the question is genuinely difficult for all students, or just difficult for certain clusters of students based on their exposure to the question, their knowledge level, and their overall educational performance.



	id	city	accuracy_level	attempt	avg_response_time	correct_answer	wrong_answer	avg_correct_answer_time	avg_wrong_answer_time
0	1269	Dhaka	0.539326	89	105.662921	48	41	111.854167	98.414634
1	652	Dhaka	0.533333	60	126.333333	32	28	137.031250	114.107143
2	273	Mymensingh	0.500000	54	128.796296	27	27	134.000000	123.592593
3	51	Khulna	0.500000	124	129.983871	62	62	139.193548	120.774194
4	1091	Dhaka	0.464789	71	114.323944	33	38	101.090909	125.815789
5	951	Barisal	0.567308	104	129.519231	59	45	123.779661	137.044444
6	70	Dhaka	0.422819	149	116.785235	63	86	108.698413	122.709302
7	839	Mymensingh	0.598639	147	118.700680	88	59	121.670455	114.271186
8	1075	Dhaka	0.571429	119	121.294118	68	51	126.926471	113.784314
9	350	Dhaka	0.509091	55	113.000000	28	27	111.821429	114.222222

Figure 3.6: Example Head of a portion of the Dataset used for our Statistical Analysis and Machine Learning Models

In addition to this, we used a set of external assumptions in order to determine difficulty level at the starting level - which includes, but is not limited to: Random Sample Consensus (Robustness Regression) of the question clusters, Theil-Sen estimator results of the answers (a generalized median-based estimator), Nearest Neighbor questions (based on a custom Ball Tree based data structure, which created nesting hyper-spheres of questions within the system), etc.

But most importantly, we used a set of basic assumption-based heuristics in order to determine the difficulty level.

For instance, with the English Questions, we are using the Dale-Chall readability index in order to determine whether students of a certain level may be served these questions. With the assumption being that students from Dhaka may be logically better at English compared to their counterparts studying from remote Villages in Bhola - given the socio-economic condition and the educational quality disparity based on regions.

Similar geographical assumptions were also automated using machine learning in order to determine whether students from a certain area (or friends, which can be detected since we have their friends' phone numbers) have a predisposition towards being able or unable to answer a certain cluster of questions, and vice versa.

For instance, students from a certain College all have their friends' phone numbers in their phone - and we can use that to design an affinity prediction mechanism resulting in a friend's-list graph/cluster. This is something that was not collected on the first version of the application, but was designed into the Second Version instead.

Given that we will be able to find their probability of being friends' using the sparsest sub-graphs based on the Collaborative Filtering (collaborative filtering methods are important and widely accepted types of recommendation systems that generate recommendations based on the ratings of like-minded users, but suffer from problems such as cold-trust, and data sparsity issues - issues which can be solved since we have the capacity to incorporate 'Trusted Information').

Thus, knowing a person, their friends, and their geological location - we will be able to predict whether students belonging to that certain cluster tend to be weaker for English. Thus, whenever a new student enters the cluster/system, we will be automatically able to understand their knowledge level, and thus serve them questions designed for their level (based on their friends' performances).

This is how the initial data cleaning was made with our basic set of data (over 14000+ questions served) in order to prepare it for the data-collection initiatives.

Similarly, topics like Mathematics and Bangla were also pre-processed to determine their difficulty levels before any real life data was achieved using those.

Additionally, our Version 1.0 of the application was launched for a short phase during the January-April 2019 Semester, within which we worked on collecting the initial test dataset based on which the rest of work has progressed.

During this time, the total dataset size reached 160,000+ individual queries processed, with over 2500 genuine users within the system. Although the version 1.0 of the app was not artificially intelligent, and did not have the capabilities that we aim to deploy within our 2.0 Application, it received a lot of positive feedback from the small community of users that we managed to reach, and did have a huge impact on our research.

It provided wide feedback and a varied dataset of users from across the country (only 63% of our users were from Dhaka, with the rest of the users organically discovering our application on the Play Store).

The dataset gained from this phase of the research is the one used for this Thesis and the experiments that we have described below.

3.3 Libraries Used

3.3.1 sckit learn

scikit-learn is a module written in Python for machine learning - which was built on top of the SciPy module.

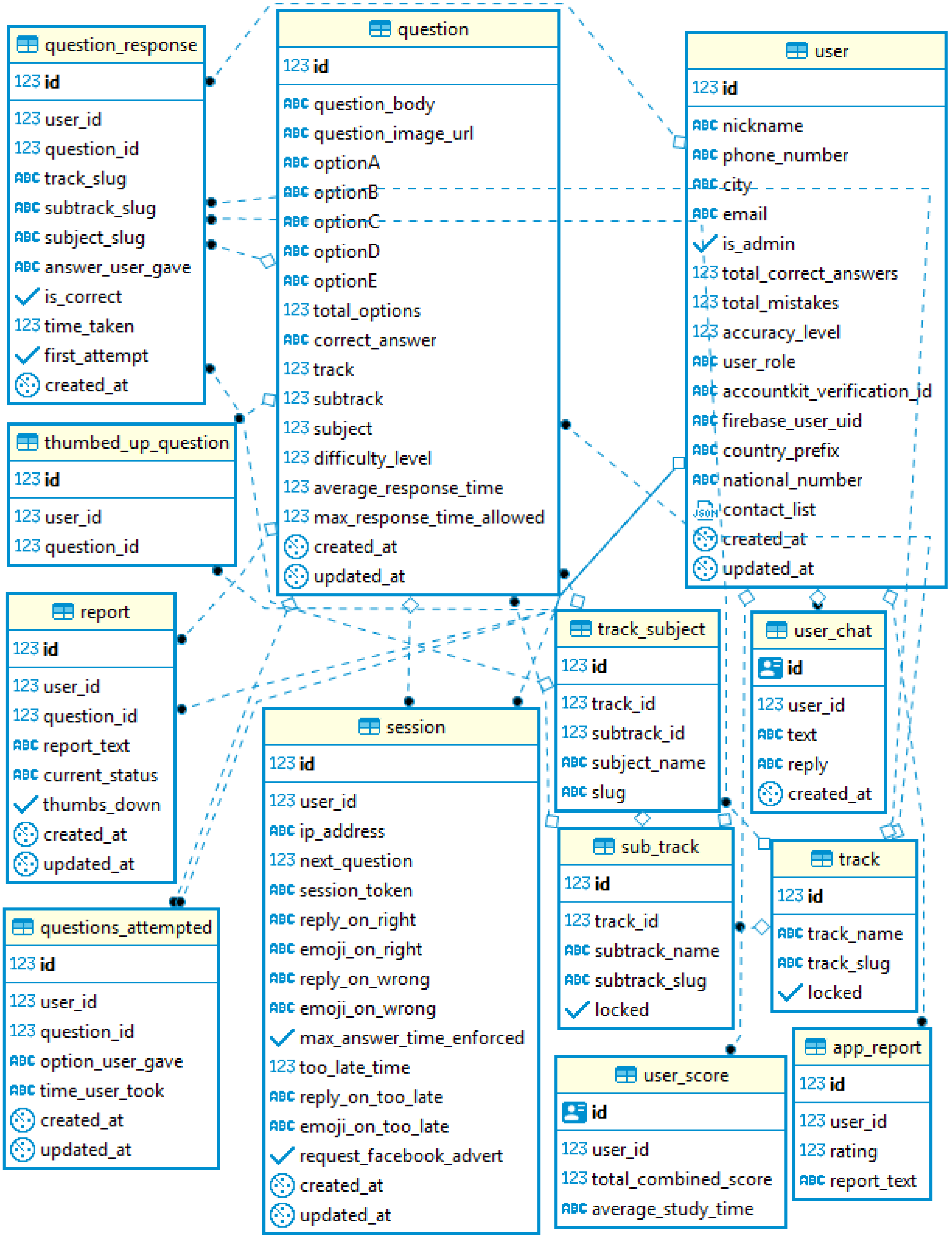


Figure 3.7: ER Diagram of the Database used on the Application Level for Data Analysis and Interpretation Modules

It features a multitude of classification, regression, and clustering algorithms - such as direct support for SVM (Support Vector Machine), Random Forests (RN), Gradient Boosting Algorithms, etc.

One of the major features of scikit-learn that we used was to do k-means clustering using this module.

3.3.2 pandas

Pandas is another software library written for the Python Programming Language, and is used for high level data manipulation and analysis.

In particular for our case, Pandas provided multiple tools and data structures and operations that allow fast and real-time data analysis, such as manipulation of numerical tables and time series automation.

For instance, Pandas Dataframes has been used widely throughout our project, and provided us with unique tools such as the ability to do pivots within the database easily - thereby assisting our statistical analysis of the dataset.

3.3.3 NumPy

NumPy is another library for Python, which allows support for large and multi-dimensional arrays and matrices, besides providing a mammoth collection of high level mathematical functions that may be operated on those arrays.

In our particular case, we used NumPy in order to operate large matrices of data that we used in our k-means algorithm through interpreting our Comma Seperated Values (CSV) into matrices that could be modified on-demand depending on our needs.

3.3.4 SciPy

SciPy is a library used for scientific computing and technical computing needs, which contains multiple modules used for linear algebra, integration, interpolation, and special functions such as signal processing.

In our specific scenario, SciPy was largely used for its ability to simply implement Algebraic Equations and carry out statistical functions, although it provides tools for k-means clustering too.

3.3.5 K-modes

The k-modes library provides a Python implementation of the k-modes and k-prototypes clustering algorithms, which also happens to heavily reply on NumPy for a majority of the functionality offered.

Since k-modes is used for clustering categorical variables, it allowed us to define clusters based on the number of matching categories between data-points (unlike the concept of clustering based on the Euclidean Distance).

3.3.6 Seaborn

Seaborn is a data visualization library for Python, which allows a high-level interface based functionality for drawing informative statistical graphics.

We used the Seaborn Library extensively throughout our work in order to generate data visualizations in order to better understand our dataset, and make interpretations of our work.

3.3.7 Matplotlib

Matplotlib is a data visualization library similar to Seaborn, but only allows a 2D plotting library that is easy to implement within the Jupyter Notebook and provides fast and simple visualization for data analysis.

Our major usage of the Matplotlib library (and the additional exploration of the PyPlot Library) was mainly concentrated on drawing necessary error charts, scatter plots, and multiple histograms based on our dataset.

Chapter 4

Methodology

4.1 Overview

On a core level, our complete solution is divided into two independent solutions that come together in order to create a complete Intelligent Teaching Platform, namely:

- Implementing a Temporally Coherent Clustering algorithm on Student Activity for Probabilistic Knowledge State Determination through the use of a complete Intelligent Teaching System, and
- Optimizing the Learning Pathway of students through a Markov Chain based Analysis of Student Performance

Overall, these two independent solutions are each sections of the whole solution that also carry multiple subsections involving Data Collection, Performance Analysis, User Clustering, Architecture Design, and Interaction Design.

All together, our framework allows for a complete system that can Predict Student Performance and then suggest an optimized learning pathway for them.

Over the next few sections, we will go over the detailed methodology that we followed in our implementation of the student performance analysis and pathway generation.

4.2 ITS Architecture

Based on the research work by Malotky, Troels and Martens, Alke (2016) [31], we concentrated on designing a singular micro-service based architecture for our Intelligent Teaching System in a concrete software engineering approach, which helped us develop a module that is capable of combining the User Modeling functionality alongside the Pedagogical/Domain Knowledge based prediction systems for a fluid experience within the Application.

We expanded upon the Architecture suggested by [31] in order to reach a design system consensus that allows us to divide the Internal Functionality of the Application into a 5 Part Microservice based Architecture, with each Microservice designed to carry out a specialized task that is vital to the overall performance of the system.

In this system, our Comparison Delivery functionality is a major functionality requested by the users, where they can see a comparison of their performance against the world/subtrack that they are competing against. The "Evaluate" functionality delivers/calculates the student's performance, and contains a major part of the clustering algorithms that we have designed for this system. The "Pathway Determination" functionality is the Markov Chain based estimator for a student's pathway determination, and allows us to determine which cluster the student needs to reach/go next.

As for the External Functionalities, the "Content" Microservice is a complete set of actions that enable us to collect and generate datasets of questions that we used in order to make the whole system work.

On the other hand, the "Domain/Expert Knowledge" Microservice is originally a set of presumptions and domain knowledge that we input externally into the system.

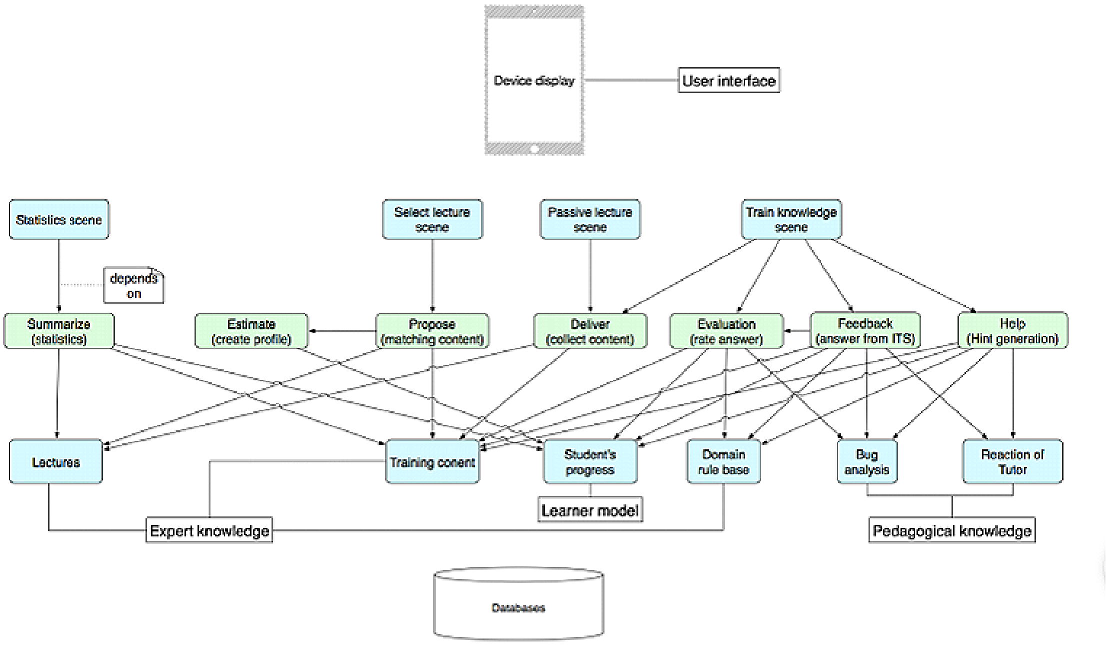


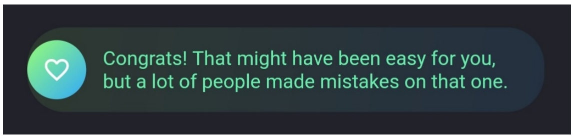
Figure 4.1: Dependencies between Functionalities within the Architecture

On the other hand, you can see the dependencies between each functionality offered by the complete ITS, based on the work from [31] and [44], which provides a complete overview of the whole system architecture and all of the moving parts within it.

4.3 Interaction Design

Given our application domain space, speech/text recognition is the most sophisticated signal that we must process at four different levels; namely, Semantic, Linguistic, Articulatory, and Acoustic.

Widely accepted as the future of interaction between human and computer interaction, our complete system will be dependent on our ability to process text data inserted by the user into the system, and to be able to interpret its intent.



Congrats! That might have been easy for you, but a lot of people made mistakes on that one.

Figure 4.2: A sample feedback within the application, detailing how an individualized conversation can take place while student is studying/taking-exam.

This will ensure that our NLU (Natural Language Understanding) backend is able to not only recognize and cluster the text, but also correctly determine its intent based on the present condition of the user (contextual meaning - which is more important than keyword recognition in our use-case).

Unlike keyword-recognition based chatbots, our contextual chatbot will be smart enough to self-improve based on what the user is asking for, and how they are asking it - while taking into consideration the overall status of their performance, and position within the system.

Our ChatBot will be based on the Amazon Lex Platform, and will leverage its NLP engine alongside our AWS Lambda Functions specifically written for our application.

As we can understand - for a conversational chatbot to be both contextually intelligent and useful, we need to follow a data-centric approach that will enable our bot to not only understand what the user wants, but also what is best for the user.

To give a use-case example of this issue: let's assume that our user is not able to answer a question, and replied with "Hint". Now, our typical contextually intelligent chatbot will carry out "keyword recognition" on this input using its pre-trained models, and automatically recognize that the user wants a hint of the current problem that is being solved.

In real life, this is an issue because typical teachers aren't supposed to give away hints to all questions - or else the whole point of taking an exam will be lost anyways.

In scenarios like this, our chatbot has to be intelligent enough to recognize and use a set of given heuristics that will enable it to make an informed decision regarding whether to serve the hint, or to serve something else altogether (such as an inspirational response, or a generic response detailing why hints will not be a good idea).

For instance, we can design custom heuristics for every grade of student, which will automatically detect the current status of the student based on those heuristics (a dimensionality reduction problem that may use Ridge Regression, an Ordinary Least Square problem where a certain penalty is imposed based on the size of the coefficients - which in our case are the results of the heuristics we have determined to be important in this scenario.

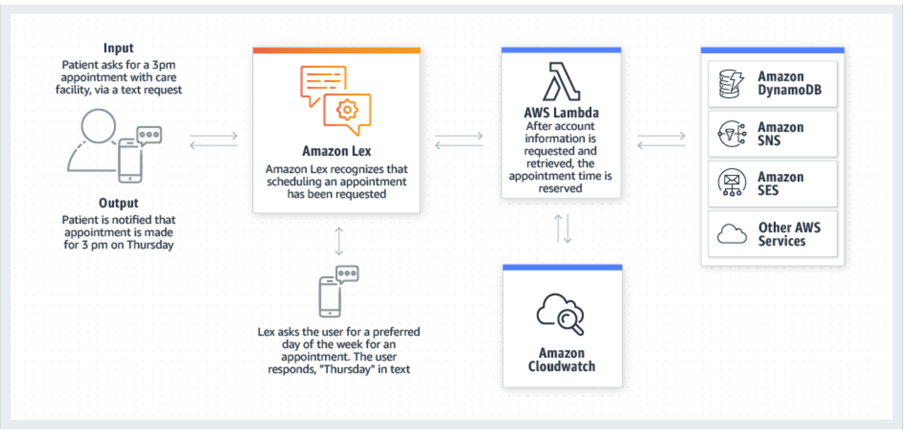

Input
Patient asks for a 3pm appointment with care facility, via a text request
Output
Patient is notified that appointment is made for 3 pm on Thursday
Amazon Lex
Amazon Lex recognizes that scheduling an appointment has been requested
Lex asks the user for a preferred day of the week for an appointment. The user responds, "Thursday" in text
AWS Lambda
After account information is requested and retrieved, the appointment time is reserved
Amazon Cloudwatch
Amazon DynamoDB
Amazon SNS
Amazon SES
Other AWS Services

Figure 4.3: Interaction of the Chatbot with the Backend Systems.

For instance, if the current context/session has served too many (weight/value to be determined dynamically using a plain Stochastic Gradient Descent learning) hints for a type of question, the system can decide that it has already served adequate numbers of hints for this type, and decide to no longer serve them a hint.

Given that Stochastic Gradient Descent for sparse data produces slightly different results than a dense implementation due to the shrunk learning rate for the intercept, we can make the logical assumption that the decision taken by the system will thus be randomized at each point, thereby ensuring that the user does not feel that the system is being too strict, or is manually coded into a decision tree algorithm.

Overall, all of these together will ensure that our ChatBot is genuinely capable of interaction in our use case, and will be able to access, authenticate, and return user data in order to be able to provide a conversational experience, and will be able to respond live to user input, thereby guiding them through their studies.

Although this is not complete, and still requires a lot of internal research and testing before it is ready for release. Since direct research behind this is not a part of our thesis proposition, we did not concentrate on this functionality so far. The system proposition is designed and talked about here for the sake of better understanding and a wide overview of the whole system and how it will perform smoothly.

Chapter 5

Implementation

5.1 Supervised Learning

5.1.1 Supervised Meta Data based Clustering

A large portion of our analysis is dependent on clustering - which involves the task of grouping similar types of students in such a way that objects in the same group (localized cluster) are more similar (in its true sense, they just share similar patterns of behaviour) to each other in the adjacent objects in the pool as compared to those in other clusters.

This is the major portion of our exploratory data mining as we invest our resources into statistical data analysis in order to recognize patterns of similar behaviour that may be used to analyze and predict their behavioral trend.

Clustering Analysis for the scope of our thesis was not contained within one specific algorithm; rather, it is achieved through the use of various algorithms and dataset manipulations - depending on the efficiency of the algorithm used.

In our specific case, since the whole algorithm needs to repeat/reiterate after every response by the user (in order to maintain the concept of a Temporally Consistent Cluster) - we decided to use k-means algorithm for a majority of the clustering requirement within our Application. This guaranteed that the process was fast/efficient enough to run at multiple con-currencies and yet not be slowed down too much that the usability of the application is affected.

Understand that since our clusters are multiple in number depending on the scenario and the dataset, it becomes a multi-objective optimization problem - and is thus an iterative process of knowledge discovery that involves trials and failures. Thus, it was often necessary for us to modify our dataset and pre-process it in order to optimize the parameters and thus achieve the desired results.

Additionally, note that since this is a concept of Temporally Inconsistent data be-

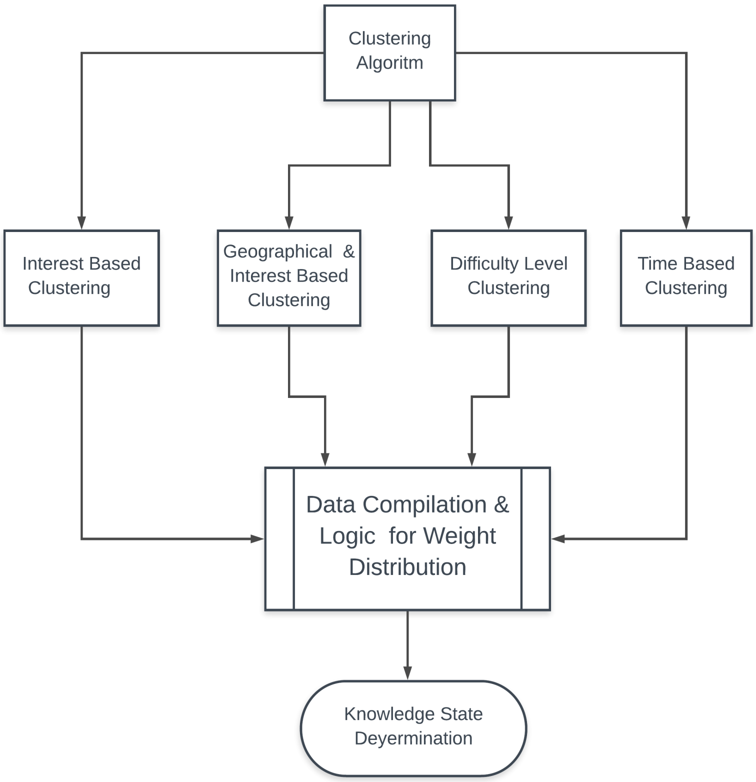

Clustering Algoritm
Interest Based Clustering
Geographical & Interest Based Clustering
Difficulty Level Clustering
Time Based Clustering
Data Compilation & Logic for Weight Distribution
Knowledge State Deyermination

Figure 5.1: Framework representing the clustering model relationships and their connections to the Application.

ing used to generate temporally consistent predictions, our cluster results thus are largely driven by domain knowledge and approximations/estimations that are manually realigned for the purpose of the research.

5.1.2 Interest Based Clustering

One of our major development bottlenecks was in the development of interest-based clustering of students and to serve questions accordingly.

We worked on developing an algorithm capable of clustering students not only based on their performance, but also on their interest in topics.

One approach to this is to use Retrieval-Based Concept Mapping, and to cluster students based on the results of that. The second, much simpler idea is to manually determine/define the interest field by defining a track/Sub-Track that the user is currently studying the most.

But over time, our question pattern revealed that Retrieval based Concept Mapping was not an optimal solution since Retrieval Based Concept Mapping is related to understanding the conceptual differences between each question, and that slowly becomes an NLP problem because concepts are usually not always differentiated using word/patterns.

For instance, the conceptual difference between two mathematical questions is not dependent on the language, but rather, the understanding of the question.

This means that we had to discard this notion, and instead proceed with cleaning our data and differentiating it into multiple different tracks/sub-tracks that would enable our users to directly enter a track that clusters them into a certain interest criteria.

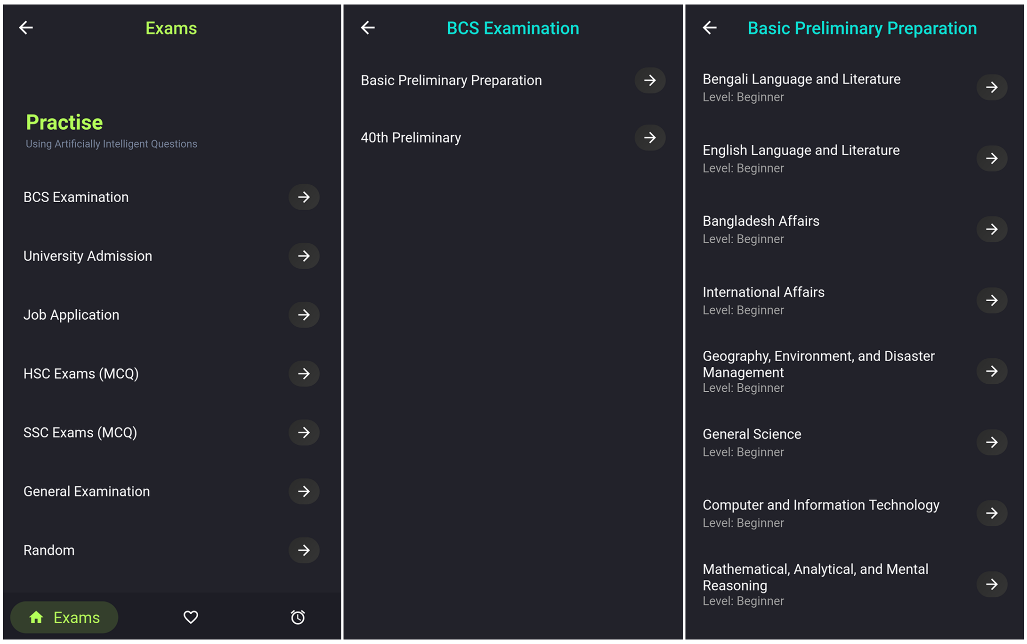


Figure 5.2: Detail of a single Track, and corresponding Sub-Track, and Subject Pages - showing how the key collections were manually identified and cleaned for proper data analysis.

This enabled us to get a better insight into the user's interest, and what can we do to serve their interest better. On a conceptual level, we can say that if we find that an individual user has over 80% of their answer attempt inside a single track/sub-track, then we can be certain that we need to optimize their experience for that interest.

Such as, if a user has mostly concentrated on studying for the BCS Exam (Track) and the 40th BCS (which was a special exam for technical hiring only), then we can easily use that cluster data in order to determine the type of additional information that we can serve him. Such as, only comparing their data with the data of related users who are attempting that same track, or have started the same track at a similar time period - thereby ensuring that data from other tracks are not polluting their performance.

Over the longer term, this ends up ensuring that a candidate who is on a single track is not being compared against the candidates who are in a different track, but are pursuing the exact same subject. These types of conflicts arised for multiple basic subjects such as English, Bangla, and Analytical Reasoning - where depending on each of the tracks, the question pattern and difficulty changes drastically, and thus cannot be considered as the same exam.

5.1.3 Geographical and Comparative Clustering

In order to prove our hypothesis that geographical challenges do have a significant impact on the result and learning abilities, we collected geographical positioning data in order to cluster students based on their geographical location, and their fellow students' performance.

We aimed to use this data in order to determine the magnitude of impact on students based on the area of residence.

This concept is derived from multiple related researches [46] [47] that suggest that there is a relationship between the geographical location and performance indicators in certain subjects, such as Science.

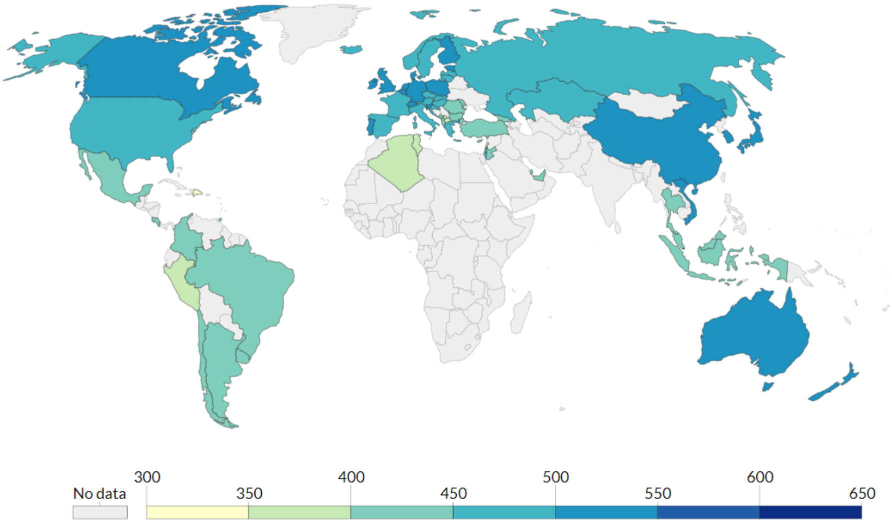


Figure 5.3: PISA test score: Mean performance on the science scale, where the metric for the overall reading scale is based on a mean for participating OECD countries set at 500, with a standard deviation of 100. Based on data from the World Bank

Although these differences are observable in the wider scale on a country level, it is also true that these studies did not reflect on the related criterion such as per-capita yearly expenditure on education, the overall education condition in the country, or even cultural differences that might have brought in such a difference.

The Figure 5.4 shows how these factors play a role in the Average Learning Outcome, and thus must be considered as an important factor when making comparisons.

Thus, we recognized the importance of these issues, and concentrated on addressing communities independently in terms of their geographical location so that optimization can be achieved on a local scale.

As an example hypothesis, we assumed that students residing in Dhaka could be better at a certain subject compared to their peers studying the exact same subject (with the exact same track).

In order to minimize the effect and create independent learning pathways that can

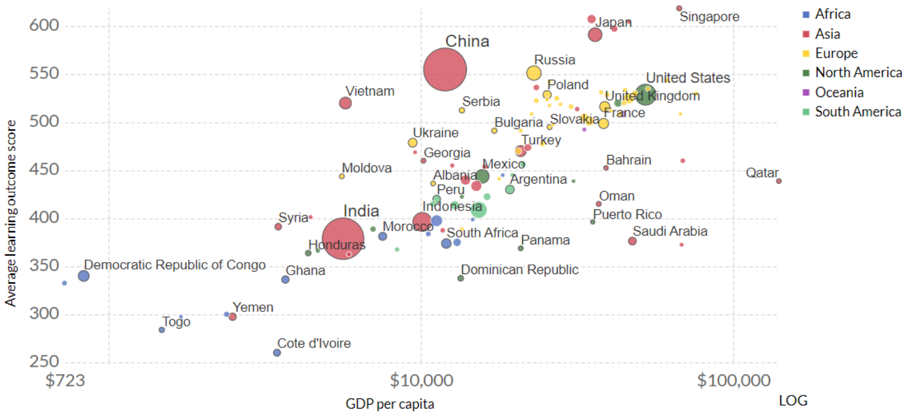


Figure 5.4: Average learning outcomes vs GDP per-capita (2015), where the vertical axis shows average scores across standardized, psychometrically-robust international and regional student achievement tests. To maximize coverage by country, tests have been harmonized and pooled across subjects (math, reading, science) and levels (primary and secondary education). The horizontal axis shows GDP per-capita after adjusting for price differences between countries and across time.

compensate for the learning and educational differences (students in Dhaka have higher access to better quality education), we optimized our algorithms in order to recognize local clusters of students and group them together first, before their learning pathway is compared against the wider geography.

On a methodical level, the challenge was to ensure that students studying from each community are not singularly placed, and the learning curve has an impact based on the total collection of students who are attempting the same subject on the same track (since they can be considered as competitors).

This was important because we needed to ensure that a Student studying Subject A in Region A is not held up at a certain Level X simply because this is above the Average of that region. Instead of hindering their learning curve from optimizing them only within their geographical division, we optimized the issue by using a Markov Chain based module which determines which stage/state to shift them to based on their current state.

5.1.4 Difficulty Level Clustering

The basic assumption behind the difficulty level based clustering is simply based on the following assumptions:

- Every Question within the ITS has its own difficulty level
- User's have a Level, which determines their current state

This is dependent on the premise that each question starts off with a single Difficulty Level. In our case, we started the Difficulty Level at 5 for each question, and then

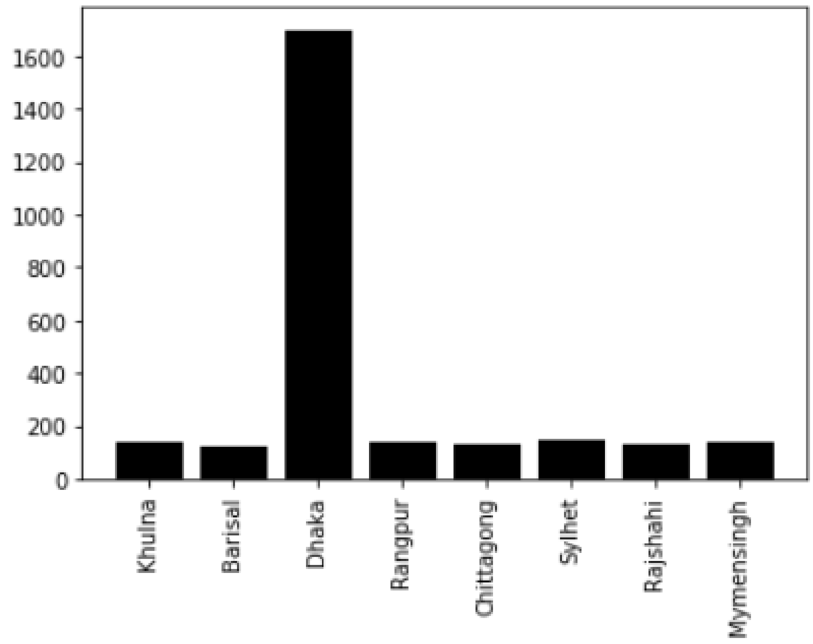


Figure 5.5: Histogram showing the Users in each Location, determined using their maximum login location during the duration of their study.

negatively/positively affected the difficulty of each question based on a few variables, namely:

- When a thresh-hold of 20 answers (n users) is reached for every question, we change the difficulty level positively/negatively based on the accuracy ratio of the student's who have answered that question.
- For questions that require a particularly large amount of time, the difficulty level is affected by the amount of time taken as compared to the current average time taken for that question.

The equation for this is expressed as:

$$D_{n=20} = AR_{n+Users} + BR + MBR + FACB \tag{5.1}$$

Where, D is Difficulty Level of the question, and AR is the Accuracy Ratio of the Question (after the latest response), and BR is the Bump Ratio, which is a rationally optimized value calculated based on factors such as the average time available for that question within that Sub-track.

If a Sub-track examination provides 100 minutes for 100 questions, then the average time available for that question is 60s. For every 80% (arbitrarily set based on domain knowledge) extra time taken over this value on an average, the Bump Ratio is upgraded by +1, due to considerations such as the time loss being a major factor for difficulty considerations.

MBR represents the Manual Bump Ratio, which is a real-life heuristsics based manual bump on a question's difficulty - based on domain knowledge and understanding of the question.

Additionally, we have included a scope for $FACB$, which is the First Attempt Correction based Bump, a manual heuristic which differentiates between questions depending on the ability of % of users who accurately answer a question on their first attempt (which means that this is a common question that is well-known within the community of users).

Through these methods, we implemented a difficulty-level clustering method, where we use the students' average accuracy ratio, first-attempt-correction ratio, time taken per type of question, and dedication level in order to come up with a holistic basic clustering of all students that is used by our question-serving algorithm. In addition to this, we also want to develop nested clusters based on the inherent difficulty of the questions based on real-life heuristics (such as, questions of Nuclear Physics are generally more difficult than questions on Magnetism).

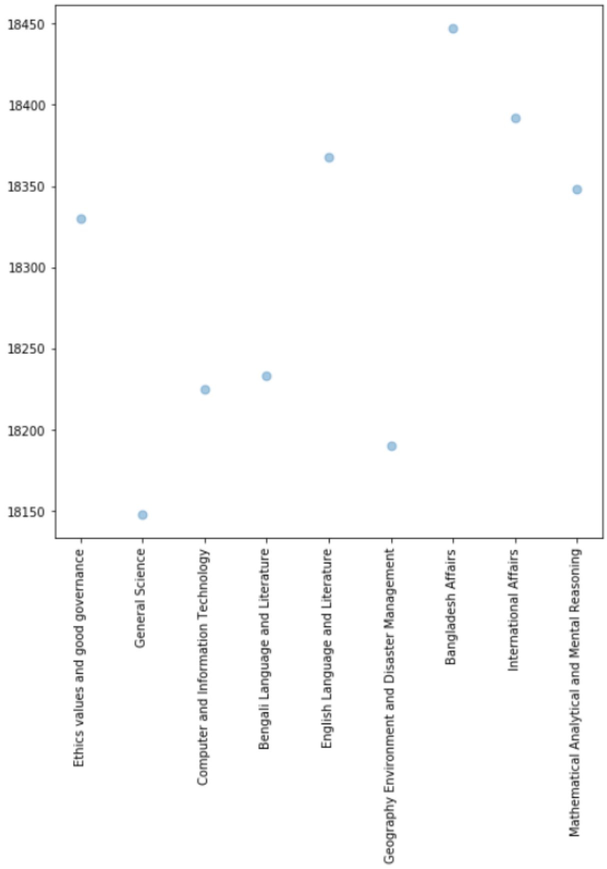


Figure 5.6: User VS Subject Scatter Plot of a Single Track, showing the number of users studying each Subject - enabling further data analysis within each datapoint

As you can see above, the scatter plot shows users who have been individually identified to have attempted questions within each subject inside a single sub-track, and thus can be exercised for the dynamic difficulty data clustering inside each subject and Sub-Track.

It is important to recognize that our difficulty level clustering is a dynamic algorithm that works inside each of the datapoints displayed above (assuming that each datapoint is a collection of users who are studying within that subject).

5.1.5 Knowledge State Determination

A large challenge with the development of the ITS is to determine the Knowledge State of each student right from their initial moment.

We determined a set of 20 test-questions during the first phase for every Subject within the available sub-track, with an even dimensional reduction algorithm employed to find the questions with the highest probability of being answered accurately at each level. Additionally, a multitude of internal heuristics were employed in order to determine which question to serve to each student. This can be best described by the equation:

$$P_{n=20} = AR_{n+Users} + BR + MBR + FACB \tag{5.2}$$

The idea is that using our set of first 20 questions, we will try to understand the student's current level of understanding. This set of 20 dynamic questions will be designed on a Subject Level, with weight of each question determined by the student's location, friends, performance of initial cluster student is in, performance of student in every question, and a multitude of other dimensions that must be addressed before we can understand what kind of questions we should serve to the student. We call this problem our Knowledge State Determination problem, and this is a one-time state-determination that is then updated regularly based on the students' performance alone in the later stages.

Overall, this allows us to make a probabilistic interpretation of the likelihood of the student being in a certain cluster - which is then re-tested by presenting questions that students from that cluster have a high probability of success at. We aim to test this hypothesis with real-life data in order to determine the accuracy of this in a production scenario.

5.1.6 Multi-Centroid Connectivity Algorithm

One of the largest issues faced while designing our complete set of algorithms was the fact that we had multiple types of information in different independent locations spreaded across the stack.

This meant that when we employed Supervised Learning Algorithms on each individual set of data, we received multiple different centroids that did not have an exact relationship between each other.

In other words, if we have the same user in two different centroids in two different k-means based clusters that were generated, then what kind of approximation/assumption can we make regarding the status of that student?

This is a major implementation hurdle because we must find a proper relationship of each of these data-points in order to make a prediction about a student's next possible step.

One suggestion to solve this problem was to assume a parametric distribution of the data, with the assumption that the sample data comes from a population which follows a definite probability distribution based on a fixed set of para-metres. Now, theoretically speaking, a normal distribution would mean that all objects in a cluster are parameterized by mean and standard deviation.

This would mean that if we know the mean, the standard deviation, and are certain that the distribution is a normal distribution, we would easily be able to find the probability of a future distribution. This would involve the construction of Mixed Gaussian Models and usage of an Expectation Maximization algorithm to predict the status of the sample in question.

But during implementation, this became an issue because there are no specific measures of accuracy available to check the result, and it would not be possible to make a judgment of the algorithm's performance without letting it loose in the real life environment and then observing its performance.

In addition to being extremely complicated to implement in terms of both time complexity and operational fit, this method also had the underlying issue that there has to be an assumption regarding the distribution of the population - and thus it would become more of goal-oriented and less of result-oriented.

But then, we do have a certain set of heuristics - which, for the sake of argument can be called "expectation measures" in the case of our algorithm.

This means that we need to find a way to maximize on our expectation regarding the performance of the clusters, or a way to make sure that the cluster performances make logical sense in terms of what we derive out of them.

This is a problem because if we consider this as an Expectation Maximization Algorithm problem - then a new issue arises that we then need to know the distribution of the new data, which in itself is a function of our parameters of our Expectation Maximization Model (EMM). Now, this can hypothetically be solved by estimating the model parameters and guessing a distribution for the new data until a convergence is achieved, but this was too complicated to perform under a low time complexity. Additionally, this could not give us any realistic results during our tests of the EMM.

Therefore, we configured our own heuristics based algorithm that employs a much simplified model that determines a weight of each of the clusters (based on multiple domain knowledge based iterations and accuracy analysis) and then generates a Cluster Score for each user:

$$CS_{user} = \sum(w * CP_{user} + (^{b}_{a}\mathrm{C})) \tag{5.3}$$

Where, CP is the Cluster Performance (or level of the individual centroid where the user belongs), multiplied by the w of each value (calculated by using an iterative method that enables the expectation maximization of our total algorithm by iteratively guessing a weight for each student until an expected result is achieved). C is the Constant that is added to the Performance depending on a (Internal Heuristics), and b (domain knowledge), and CS is the Cluster Score of the User, which is then used in the question generation algorithm (simply put, the higher the cluster score, the better a student is - and the higher they need to go - and depending on their gradient ascent or decent, upgraded to the next level of questions).

5.2 Prediction & Interpretation

As a part of our ITS, we aim to provide prediction services such as answers to whether a student will pass an exam or not, and help our question serving logic with that data in order to determine what types of questions should be served next.

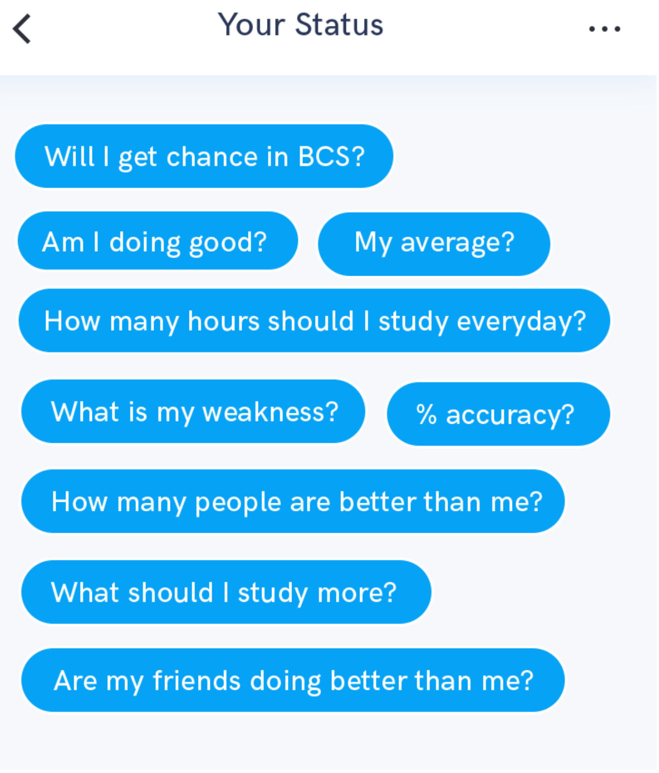

Your Status
Will I get chance in BCS?
Am I doing good?
My average?
How many hours should I study everyday?
What is my weakness?
% accuracy?
How many people are better than me?
What should I study more?
Are my friends doing better than me?

Figure 5.7: Prediction Requests that the user can make to the system!.

This data is processed and achieved by using the previously mentioned temporally coherent clustering of our students, and then employing the Multi Centroid Connectivity Algorithm's gradient ascent or decent in order to reach an estimation regarding the future performance.

For instance, for the question “Will I get chance in ¡track¿ Examinations”, we take into account not only the student’s learning curve, but also the learning curve of all students studying that subject due to the dynamic nature of our system. Thus, we proactively determine a pass-level for that subject for the coming year (based on the performance of the students in the current year, and the acceptance ratio of the next exam) - and then determine whether their current performance will be adequate to help them gain a place in the top cluster of students.

For questions like, “How many hours should I study everyday”, we use the student’s current learning curve and used a linear regression analysis in order to determine whether their accuracy level will rise up adequately within the time-frame left before the exam (gradient ascent approximation).

Very simply put, if the gradient of their rise is not promising enough to guarantee a “success” in the exam, we will cross-relate that data with their current hours, and raise the bar to a level where it should give a better output - with the heuristic assumption based on the probability that given a certain amount of dedication per day within studies, results are bound to increase up-to a certain level.

In one prototype of this system, we used unsupervised birch-algorithm in order to implement an effective single-scan, scalable clustering on our multi-dimensional metric data points of the student performance in an attempt to produce the best quality hierarchical clustering with minimal resource consumption. But, we need to test this implementation on real-life data in order to determine its performance on the field.

In our initial test data set, we have achieved upto 84% accuracy on our data for this operation.

Chapter 6

Results

Throughout our work, we concentrated on an evolutionary pipeline for clustering of student behavior. Since this was designed for one specific kind of situation in mind, the tests and the accuracy that it provides are tested for that result only.

We demonstrated that enforcing a temporal coherency between two consecutive clustering through using the Markov Model is beneficial for the detection of student behaviour, and as well as for detection for *cluster events*.

By applying our pipeline to the action sequences generated by our test dataset, we revealed interesting properties regarding the behavior of students and potential deficiencies within current learning environments.

The figure below shows the relative cluster centers (centroids) we achieved for our dataset, with an accuracy/trust of 84%.

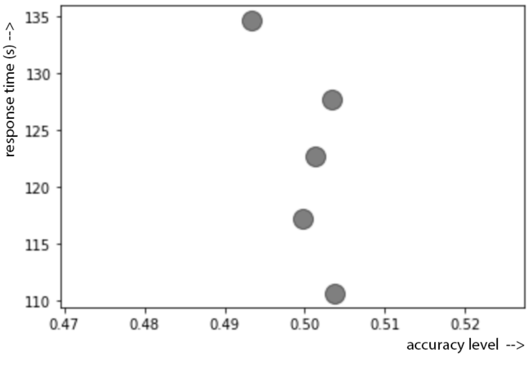


Figure 6.1: Centroid Count and Placement Display achieved by the Clustering Algorithm based on k-means. Based on Response Time VS Accuracy Levels.

The optimal number of cluster centroids was determined using the Elbow Method, which is a method that provides the best idea on what a good k number of clusters would be. This is determined by using the sum of the squared distance between data points and their assigned cluster centroids.

As you can see on the image 6.1 picked the spot where the Sum of the Squared

Distances starts to flatten out and form an elbow, while giving the best possible accuracy ratio. The reason why we did not choose higher number of clusters (given that they do increase accuracy ratio) is because then it poses the risk of over-fitting the dataset.

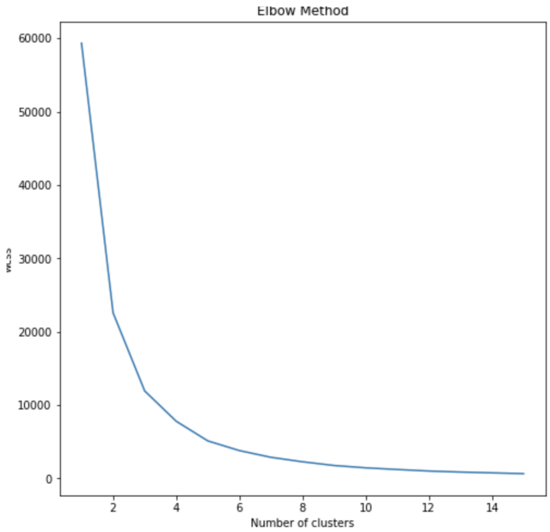
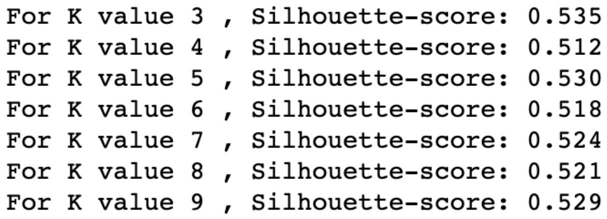


Figure 6.2: Elbow method Analysis in order to determine Optimal k Value

Additionally, the silhouette data shows the accuracy ratio that we managed to achieve.



```
For K value 3 , Silhouette-score: 0.535
For K value 4 , Silhouette-score: 0.512
For K value 5 , Silhouette-score: 0.530
For K value 6 , Silhouette-score: 0.518
For K value 7 , Silhouette-score: 0.524
For K value 8 , Silhouette-score: 0.521
For K value 9 , Silhouette-score: 0.529
```

Figure 6.3: Silhouette Score of the Clustering Algorithm based on k-means, showing the Accuracy Ratio Achieved

Since in real life implementation, this student behavior will be clustered continuously over multiple sessions, it means that the cluster number and the cluster sizes will definitely change over time. We expect the clusters to merge, split, dissolve, and form based on the then-current dataset condition and the performance analysis.

In a simulated example of four types of cluster events, research at the ETM Zurich [1] revealed that the cluster behaves in the expected reasoning using synthetic data in an artificial scenario.

For instance, it was investigated that a Group A consisting of bad performing students (Dark Green) merges into group B (Dark Blue) of good students in the top left image above, they both start using the "help" data within the ITS. Which was expected.

Similarly, the other expected behaviors were also noticed throughout the experiments carried out. Additionally, in order to show the shift of students from one

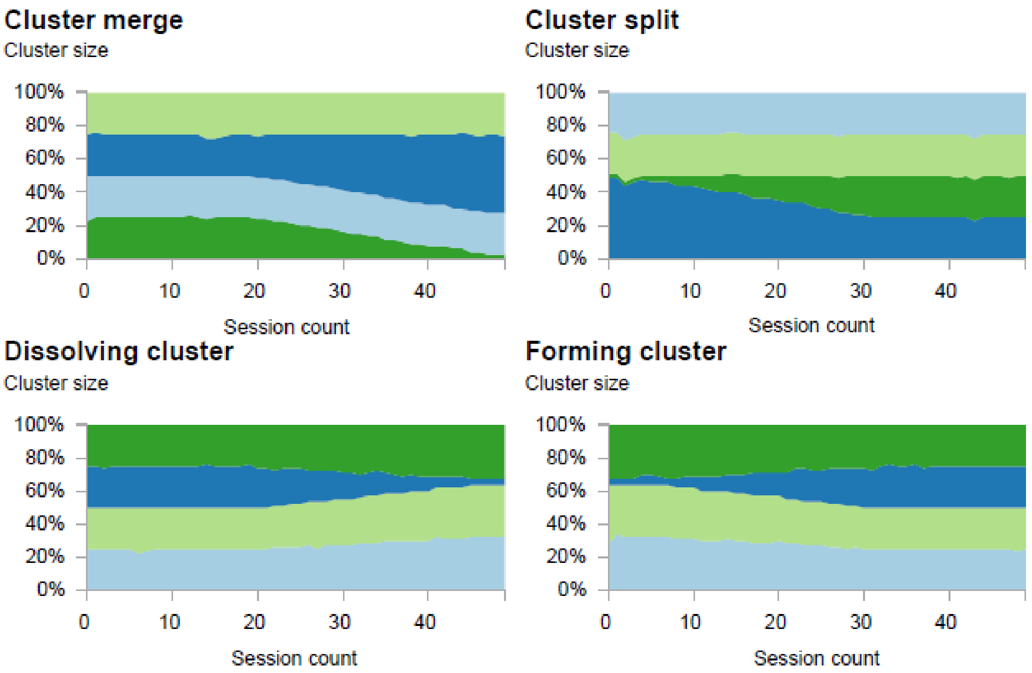


Figure 6.4: Expected future cluster behavior based on Simulated Behavior

cluster to another cluster, we applied our Markov Chain Analysis on the current dataset we generated using our Version 1.0 of the Application.

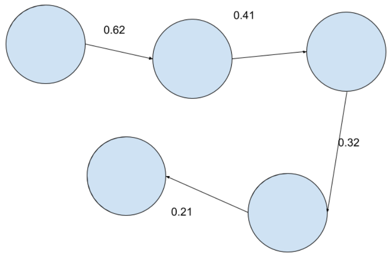


Figure 6.5: Markov Chain showing Probability of Cluster Shifting based on Student Behavior

This revealed an expected behaviour (given that we are only considering movement forward and not backwards) of the students gradually having a lesser percentage probability of "graduating" to the next cluster level (which is the tougher difficulty level based cluster).

We used Markov models to classify and compare the performance of students, who are originally in one of the "Levels" based on a steady state reached within our assessment.The Levels and our associated Accuracy Ratio's are listed on the table 6.1.

Level	**Score Classification**
Level 1	0.1
Level 2	0.2
Level 3	0.3
Level 4	0.4
Level 5	0.5 (start)
Level 6	0.6
Level 7	0.7
Level 8	0.8
Level 9	0.9
Level 10	1.0

Table 6.1: Accuracy Ratio VS Level of Student Performance

Ideally, any student can navigate across our level algorithm in a gradual manner, which can be characterized as a series of random variables for each student, which represents the state of the student within the system at SL_1, SL_2, SL_3 ... SL_{10}. A student begins our system at a level k determined dynamically based on the amount of data available to us, but k is kept constant at a Level 5 for all ”new” students within the system. But this is not always true because for select sub-tracks, we served students with a certain set of pre-set questions that enable us to “guesstimate” their accuracy ratio, and accordingly start them in a certain state. The number of questions that we needed to serve them was decided dynamically, given that until we found an 80% accuracy ratio in a certain Difficulty Level, it is not possible to set this state, and the constraint is set in that manner.

The idea is that student can either move to SL_{k+1} or SL_{k-1} at each step depending on the change in their accuracy ratio and their score at that level, but it is not possible for them to move to any other level (this maintains the principle of the Markov Chain because each state depends only on the previous state. The transitions between these states can be seen as a Markov Chain and the transition probability table shown below:

$$\begin{array}{c} \\ SL_1 \\ SL_2 \\ SL_3 \\ SL_4 \\ SL_5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} SL_1 & SL_2 & SL_3 & SL_4 & SL_5 \end{array} \\
\begin{pmatrix}
P_{11} & P_{12} & 0 & 0 & 0 \\
P_{21} & 0 & P_{23} & 0 & 0 \\
0 & P_{32} & 0 & P_{34} & 0 \\
0 & 0 & P_{43} & 0 & P_{45} \\
0 & 0 & 0 & P_{54} & P_{55}
\end{pmatrix}
\end{array}$$

The initial state transition probability for adaptive assessment of new students is shown in the table below:

Note that while this data is true for a single student within our given dataset, it is widely varying for each scenario depending on the student for whom we are developing this matrix.

The Table 6.2 shows us the percentage of the students who started the test at each of our levels, which displays a formation very similar to a typical bell formation in the graph that is typical of a large student dataset, albeit with some inclinations since the dataset was not manually forced to follow a curve pattern.

Level	**Percentage of Students**
Level 1	2%
Level 2	5%
Level 3	8%
Level 4	13%
Level 5	26%
Level 6	16%
Level 7	11%
Level 8	10%
Level 9	6%
Level 10	3%

Table 6.2: Percentage of "New" Students who started at each level

This shows that our initial algorithmic state analysis allowed us to obtain an expected ratio of state levels that are evenly distributed without requiring pre-specified distribution, which guarantees uniform performance across future data-sets.

While it is true that it is not possible to test our Markov Chain based pathway generation algorithm without deploying it in real life and collecting further student data, we can make an assumption from these results that the concept is hypothetically concrete, and gives us expected results in the simulated scenario.

This also means that the results regarding this portion of the work are not incomplete, but rather, stalled due to lack of further work in this arena. Given the fact that Steady State probabilities achieved using the Markov Chain based analysis has proved to be largely successful by the work of previous researchers [1], we can say that it is already a proven model that is not questioned.

Overall, our complete set of results reflect our expectations from this thesis work, and are representative in the sense that they did not display any bias-ness or any irregularity that is beyond explanation. This proves that our hypothesis is well-grounded in its own merit, and hopes to make the ends meet in the long run.

Chapter 7

Future work And Conclusions

Overall, our thesis was divided into two major parts - namely:

- Implementation of a Temporally Coherent Clustering for the prediction of student results
- Using the Markov Chain Model in order to generate the best pathway for learning curve optimization

Ideally, the best scenario would have been a fully stable and hypothetically irrefutable thesis where our work has been completely tested beyond question. But, due to certain limitations such as lack of data-sets larger than what we have, unique data requirements that are not available in public data-sets worldwide, and inability to test the algorithm's work in the real world - they combined in order to give us much scope of future work in this field.

This also means that we still have the scope to improve on our models, and increase their accuracy. For instance, using k-means/k-mode algorithms for our clustering model may be fast (which was an important requirement) and efficient (in terms of accuracy achieved), but it is definitely a naive approach towards solving a complex problem of this magnitude. In the future, it would be wise for us to implement and test further clustering algorithms in order to increase the "trust" factor within our work.

Additionally, much of our work in terms of the complete application is yet to be completed. Such as research (unrelated to this Thesis) regarding Natural Language Cognition for better understanding the answers of the students in the context of our application domain, further research into the usability and practicality of our pathway generation algorithm in the context beyond MCQ questions, etc.

Also, one of the major improvements that we can bring within our models is to get rid of manual heuristics and weights within our algorithm - and slowly shift towards a neural network based approach, which would solve the problem of biasness when it comes to applying domain knowledge into the algorithm. This needs to be explored further at its own pace, and will definitely drive a new arena of accuracy and learning in our field of research.

We hope that this Thesis will open up a pathway to many similar initiatives and work in this field, and we aim to continue research this arena alongside our future studies.

Bibliography

[1] S. Klingler, B. Solenthaler, T. Käser, and M. Gross, “Temporally coherent clustering of student data”, *Computers and Society*, Jun. 2017. [Online]. Available: http://www.educationaldatamining.org/EDM2016/proceedings/paper_11.pdf.

[2] B. Solenthaler, S. Klingler, T. Käser, and M. Gross, “Ten years of research on intelligent educational games for learning spelling and mathematics”, *Computers and Society*, Jun. 2018. [Online]. Available: https://arxiv.org/pdf/1806.03257.pdf.

[3] C. Hansen, C. Hansen, N. Hjuler, S. Alstrup, and C. Lioma, “Sequence modelling for analysing student interaction with educational systems”, *Computers and Society*, Jan. 2018. [Online]. Available: https://www.researchgate.net/publication/319122146_Sequence_Modelling_For_Analysing_Student_Interaction_with_Educational_Systems.

[4] N. Patel, C. Sellman, and D. Lomas, “Mining frequent learning pathways from a large educational dataset”, *Computers and Society*, Jul. 2017. [Online]. Available: https://arxiv.org/abs/1705.11125v3.

[5] S. B. H. S. B. Ravindran, “Predicting students’ performance on intelligent tutoring system — personalized clustered bkt (pc-bkt) model”, vol. 241, pp. 65–78, 2007.

[6] P. Nedungadi and M. S. Remya, “Mining frequent learning pathways from a large educational dataset”, *Computers and Society*, Oct. 2014. [Online]. Available: https://ieeexplore.ieee.org/document/7044200.

[7] I. Jugol, B. Kovačić1, and E. Tijan, “Cluster analysis of student activity in a web-based intelligent tutoring system”, *Multidisciplinary Scientific Journal of Maritime Research*, Jun. 2015.

[8] B. Kovačić1, I. Jugol, and E. Tijan, “Mining student behavior patterns in reading comprehension tasks.”, *Educational Data Mining*, 2012.

[9] J. K. D. Perera, I. Koprinska, K. Yacef, and O. R.Zaıane, “Clustering and sequential pattern mining of online collaborative learning data”, *Transactions on Knowledge and Data Engineering*, 2009.

[10] R. Martinez-Maldonado, K. Yacef, and J. Kay, “Data mining in the classroom: Discovering groups’ strategies at a multi-tabletop environment”, *Educational Data Mining*, 2013.

[11] Y. Wang and N. T. Heffernan, “The student skill model”, *nternational Conference on Intelligent Tutoring Systems*, 2012.

[12] Y. Bergner, Z. Shu, and A. A. V. Davier, “Visualization and confirmatory clustering of sequence data from a simulation-based assessment task”, *Educational Data Mining*, 2014.

[13] J. Herold, A. Zundel, and T. F. Stahovich, “Mining meaningful patterns from students’ handwritten coursework”, *Educational Data Mining*, 2013.

[14] L. D. Miller and L.-K. Soh, “Meta-reasoning algorithm for improving analysis of student interactions with learning objects using supervised learning”, *Educational Data Mining*, 2013.

[15] J. M. L. Andres, M. M. T. Rodrigo, R. S. Baker, L. Paquette, V. J. Shute, and M. Ventura, “Analyzing student action sequences and affect while playing physics playground”, *Educational Data Mining*, 2015.

[16] J. S. Kinnebrew and G. Biswas, “Identifying learning behaviors by contextualizing differential sequence mining with action features and performance evolution”, *Educational Data Mining*, 2012.

[17] R. Agrawal and R. Srikant, “Mining sequential patterns. in data engineering”, *IEEE*, 1995.

[18] R. Martinez-Maldonado, K. Yacef, and J. Kay, “Data mining in the classroom: Discovering groups’ strategies at a multi-tabletop environment”, *Educational Data Mining*, 2013.

[19] G. Biswas, H. Jeong, J. S. Kinnebrew, B. Sulcer, and R. ROSCOE, “Measuring self-regulated learning skills through social interactions in a teachable agent environment”, *Research and Practice in Technology Enhanced Learning (RPTEL)*, 2010.

[20] K. E. Boyer, R. Phillips, E. Y. H. A. Ingram, M. Wallis, M. Vouk, and J. Lester, “Characterizing the effectiveness of tutorial dialogue with hidden markov models”, *International Conference on Intelligent Tutoring Systems*, 2000.

[21] J. S. Kinnebrew and G. Biswas, “Identifying learning behaviors by contextualizing differential sequence mining with action features and performance evolution”, *Educational Data Mining*, 2012.

[22] M. Desmarais and F. Lemieux, “Clustering and visualizing study state sequences”, *Educational Data Mining*, 2013.

[23] M. Kock and A. Paramythis, “Activity sequence modelling and dynamic clustering for personalized e-learning”, *User Modeling and User-Adapted Interaction (UMUAI)*, 2011.

[24] C. Li and G. Biswas, “A bayesian approach to temporal data clustering using hidden markov models”, *International Conference on Machine Learning*, 2000.

[25] J. S. Kinnebrew, D. L. Mack, and G. Biswas, “Mining temporally-interesting learning behavior patterns”, *Educational Data Mining*, 2013.

[26] D. Chakrabarti, R. Kumar, and A. Tomkins, “Evolutionary clustering”, *Knowledge Discovery in Databases*, 2006.

[27] K. S. Xu, M. Kliger, and A. O. H. Iii, “Adaptive evolutionary clustering. data mining and knowledge discovery”, 2014.

[28] K. Aggarwal, N. Aggarwal, and P. Makkar, “Analysis of k-means clustering algorithm for data mining”, *national conference on emerging trends in electronics and information technology*, 2012.

[29] R. loohach and D. K. Garg, “An insight onerview of issuses and challenges associated with clustering algorithms”, *International Journal of Research in IT and Management (IJRIM)*, 2012.

[30] R. Jingbiao and Y. Shaohong, ““research and improvement of clustering algorithm in data mining”, *International Conference on Signal Processing Systems*, 2010.

[31] N. Troels, G. von Malotky, and I. A. Martens, “Framework for intelligent teaching and training system – a study of system”, *University of Rostock-Institute of Computer Science*, 2016.

[32] S. D.J., C. M., S. B., and S. E.S., “Student engagement in high school classrooms from the perspective of flow theory”, *Springer, Dordrecht*, 2014.

[33] C. T. and A. B., “Spatio-temporal clustering of road network data”, *Artificial Intelligence and Computational Intelligence*, 2010.

[34] M. VL, G. JC, and S. JP, “Temporal clustering analysis: What does it tell us about the resting state of the brain?.”, *Med Sci Monit*, 2008.

[35] Koedinger, K. R. Anderson, J. R. Hadley, W. H., and M. A. Mark, “Intelligent tutoring goes to school in the big city”, *Journal of Artificial Intelligence in Education*, 1997.

[36] Corbett, A. Anderson, J. Graesser, A. Koedinger, and V. K, “Generation computer tutors: Learn from or ignore human tutors?”, *In Proceedings of the 1999 Conference of Computer-Human Interaction*, 1999.

[37] R. Guo, D. Palmer-Brown, S. W. Lee, and F. F. Cai, “Intelligent diagnostic feedback for online multiple-choice questions”, *Artificial Intelligence Review*, 2014. [Online]. Available: https://doi.org/10.1007/s10462-013-9419-6.

[38] T. K. Dey, A. Rossi, and A. Sidiropoulos, “Temporal clustering”, *Computers and Society*, Apr. 2017. [Online]. Available: https://arxiv.org/pdf/1704.05964.pdf.

[39] B. M. Varghese, J. T. J., U. A., and P. J. K, “Clustering student data to characterize performance patterns”, *International Journal of Advanced Computer Science and Applications*, 2013. [Online]. Available: https://thesai.org/Downloads/SpecialIssueNo3/Paper.

[40] M. Ester, A. Frommelt, H.-P. Kriegel, and J. Sander, “Algorithms for characterization and trend detection in spatial databases”, *Conf. on Knowledge Discovery and Data Mining*, 1998.

[41] A. Frommelt, M. Ester, H.-P. Kriegel, and J. Sander, “Algorithms for characterization and trend detection in spatial databases”, *Conf. on Knowledge Discovery and Data Mining*, 1998.

[42] L. Kaufman and P. J. Rousseeuw, “Finding groups in data: An introduction to cluster analysis”, *Wiley-Interscience*, 2005.

[43] S. P. Borgavakar and A. Shrivastava, "Evaluating student's performance using k-means clustering", *International Journal of Engineering Research Technology*, 2017.

[44] Z. Zhu, Y. Sun, and P. Riezebos, "Introducing the smart education framework: Core elements for successful learning in a digital world", *International Journal of Smart Technology and Learning*, vol. 1, p. 53, Jan. 2016. DOI: 10.1504/IJSMARTTL.2016.078159.

[45] Z. Fan, Y. Sun, and H. Luo, "Clustering of college students based on improved k-means algorithm", *Journal of Computers*, 2016. DOI: 10.3966/1991159920171.

[46] EdStats. (). Ed stats from world bank, [Online]. Available: https://datacatalog.worldbank.org/dataset/education-statistics. (accessed: 02.08.2019).

[47] Altinok, N. Angrist, N. Patrinos, and H. Anthony. (). Global data set on education quality (1965-2015), [Online]. Available: http://documents.worldbank.org/curated/en/706141516721172989/Global-data-set-on-education-quality-1965-2015. (accessed: 23.01.2018).

[48] A. de Acosta, "Large deviations for empirical measures of markov chains", *Journal of Theoretical Probability*, vol. 3, no. 3, pp. 395–431, Jul. 1990, ISSN: 1572-9230. DOI: 10.1007/BF01061260. [Online]. Available: https://doi.org/10.1007/BF01061260.

[49] A. Tesselkin and V. Khabarov, "Estimation of origin-destination matrices based on markov chains", *Procedia Engineering*, vol. 3, no. 3, pp. 395–431, Jul. 2017, ISSN: 1572-9230. DOI: 10.1016/j.proeng.2017.01.071. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877705817300711.

[50] P. Rai and A. Lal, "Google pagerank algorithm: Markov chain model and hidden markov model", *International Journal of Computer Applications*, Mar. 2016. [Online]. Available: https://pdfs.semanticscholar.org/3eb3/40d9605a6e%2046d49302f0be945afcff43da1b.pdf.

[51] C. Piccardi, "Finding and testing network communities by lumped markov chains", *PLOS ONE*, vol. 6, pp. 1–13, Nov. 2011. DOI: 10.1371/journal.pone.0027028. [Online]. Available: https://doi.org/10.1371/journal.pone.0027028.

[52] M. J. O. P. Miron F. J. Beron-Vera and P. Koltai, "Markov-chain-inspired search for mh370", *Chaos: An Interdisciplinary Journal of Nonlinear Science*, Mar. 2019. DOI: 10.1063/1.5092132. [Online]. Available: https://doi.org/10.1063/1.5092132.

[53] G. P. J. Varadraj P. Gurupur and R. Rudraraju, "Evaluating student learning using concept maps and markov chains", *Expert Systems with Applications*, 2015. DOI: 0.1016/j.eswa.2014.12.016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417414007945.

[54] C. MAVRUK and E. KIRAL, "Academic progress of students in quantitative courses at nigde university vocational school of social sciences: A prediction using markov model", *Niğde Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 2016. [Online]. Available: https://dergipark.org.tr/download/article-file/226567.

[55] R. Hlavatý and L. Dömeová, “Students’ progress throughout examination process as a markov chain”, *International Education Studies*, Nov. 2014. [Online]. Available: http://dx.doi.org/10.5539/ies.v7n12p20.

[56] V.Geetha, G.S.Nandakumar, and R.Vivek, “Application of markov chain for comparing the performance of students in adaptive e-assessment”, *International Journal of Pure and Applied Mathematics*, 2018. [Online]. Available: https://acadpubl.eu/jsi/2018-118-18/articles/18c/61.pdf.

[57] A. Brezavšček, M. Pejić Bach, and A. Baggia, “Markov analysis of students’ performance and academic progress in higher education”, *Organizacija*, vol. 50, pp. 83–96, May 2017. DOI: 10.1515/orga-2017-0006.

[58] A.-L. Wang, “Introducing markov chains models to undergraduates”, *International Statistical Institute*, 2001. [Online]. Available: https://iase-web.org/documents/papers/isi53/596.pdf.