Enhancing Personalized Learning of students through Deep Learning in an Adaptive Learning Environment

A dissertation submitted for the Degree of Master of Business Analytics

M.P.M.I.Pathirana

University of Colombo School of Computing

2021

Declaration

Name of the student: M.P. Madushan Isuru Pathirana
Registration number: 2020BA04
Name of the Degree Programme: Master of Business Analytics
Project/Thesis title: Enhancing Personalized Learning of Students through Deep Learning in an Adaptive Learning Environment

1. The project/thesis is my original work and has not been submitted previously for a degree at this or any other University/Institute. To the best of my knowledge, it does not contain any material published or written by another person, except as acknowledged in the text.

2. I understand what plagiarism is, the various types of plagiarism, how to avoid it, what my resources are, who can help me if I am unsure about a research or plagiarism issue, as well as what the consequences are at University of Colombo School of Computing (UCSC) for plagiarism.

3. I understand that ignorance is not an excuse for plagiarism and that I am responsible for clarifying, asking questions and utilizing all available resources in order to educate myself and prevent myself from plagiarizing.

4. I am also aware of the dangers of using online plagiarism checkers and sites that offer essays for sale. I understand that if I use these resources, I am solely responsible for the consequences of my actions.

5. I assure that any work I submit with my name on it will reflect my own ideas and effort. I will properly cite all material that is not my own.

6. I understand that there is no acceptable excuse for committing plagiarism and that doing so is a violation of the Student Code of Conduct.

Signature of the Student	**Date (DD/MM/YYYY)**
	29/09/2024

Certified by Supervisor(s)

This is to certify that this project/thesis is based on the work of the above-mentioned student under my/our supervision. The thesis has been prepared according to the format stipulated and is of an acceptable standard.

	Supervisor 1
Name	Dr. L. N. C. De Silva
Signature	
Date	29 / 09 / 2024

I would like to dedicate this thesis to my parents and family who have continuously supported me to come to the stage where I am today.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my research supervisor, Dr. Lasanthi De Silva, for their invaluable guidance, unwavering support, and insightful feedback throughout the journey of this thesis. Their expertise and encouragement have been instrumental in shaping this research and enhancing its quality.

I am also deeply grateful to my employer, John Wiley & Sons, for their generous support in providing access to the necessary data and resources essential for the completion of this study. Their cooperation and assistance have significantly contributed to the success of this research endeavor.

I extend my heartfelt thanks to all those who have directly or indirectly contributed to this thesis, including my family and friends, for their encouragement and understanding during this challenging yet rewarding academic pursuit.

ABSTRACT

Adaptive learning and online education have gained significant importance in recent years. In this study, we apply Graph Neural Network (GNN) and Recurrent Neural Network (RNN) techniques to the task of Knowledge Tracing. Unlike previous literature that necessitates searching for the most relevant questions, our methodology focuses on the utilization of the most recent questions from the exercise history. Additionally, while prior studies have employed bidirectional graphs to incorporate question information and learning objectives, our model constructs directional graphs that consider the hierarchy of learning objectives. This hierarchical structure guides the propagation of question and learning objective embeddings, enabling a more contextually informed representation of student knowledge. We compare the performance of our model with question embeddings to a model without, revealing that the incorporation of question embeddings significantly enhances predictive accuracy. Our findings underscore the importance of adaptive learning methodologies in online education, offering insights into more effective knowledge assessment and personalized learning experiences.

Table of Contents

Table of Figures

1 CHAPTER 1 - INTRODUCTION

1.1 Introduction to E-learning

Education is one of the fundamental pillars in a society that drives intellectual growth and uplifts social standards. According to United Nations, Universal Declaration of Human Rights, Article 26, ‘Everyone has a right to education’ (UN General Assembly, 1948). At the beginning of the last century, education focused on knowledge and skills without considering the learner's expectations and learners abilities. Hence the ‘one size fits all’ education system faced challenges in catering to individual student requirements. Personalized teaching and learning frameworks immerged to fill this gap with the development of technology. Learning Management Systems (LMS), Adaptive Hypermedia Systems (AHS), and Intelligent Tutoring Systems (ITS) are to name a few systems developed to cater to personalized education. (Katsaris & Vidakis, 2021). Table 1.1 further explain each E-learning system.

Table 1.1 Types of E learning systems

E-learning systems	Characteristics
Learning Management Systems	LMS delivers content and help administrative tasks
Adaptive Hypermedia Systems	Provide content based on user goal and performance
Learning Style based Adaptive Educational Systems	Personalize the learning experience based on learning style (visual, auditory, reading/writing, and kinesthetic)
Intelligent Tutoring Systems	Provide immediate and customized instruction/feedback without human intervention using Adaptive Learning

This study focues in Intelligent Tutoring Systems. According to (Mousavinasab et al., 2021) Intelligent Tutoring Systems (ITSs) consist of four main modules. The first is the expert module, containing domain knowledge and problem-solving techniques. The second is the student diagnosis module, which gathers and updates information about the learner's knowledge, activities, and responses. The third is the instruction module, which detects knowledge deficiencies and employs teaching strategies to address them using adaptive learning technologies. The last module is the user interface, facilitating communication between the user and the system. Incorporating

AI techniques, e-learning systems have aimed to enhance adaptive and customized learning. Adaptive feedbakcs what makes intelligent tutoring systems really intelligent. This study further focues on the third instrudction models' adaptive learning capabilities and how to improve adaptive learning process using deep learning.

1.2 Introduction to adaptive learning

Adaptive learning is a methodology for teaching and learning that strives to personalize lessons, readings, practice activities, and assessments for individual students based on their current skills and performance. Adaptive learning systems use a data-driven approach to adjust the path and pace of learning, enabling the delivery of personalized learning at scale (Ennouamani & Mahani, 2018).

Adaptive learning is a type of scaffolding technique used in educational technology that is tailored to support all stakeholders in an educational institution, including teachers, students, and school administrators. According to (Jan-Martin Lowendahl et al., 2016) adaptive learning adjusts instructional content based on student responses and preferences, relying on learning data and algorithmic pedagogical responses.

1.2.1 Importance of adaptive learning

There are many benefits of adaptive learning. Adaptive learning saves teachers time and provides data and analytics that help to understand students. For students, it provides a personalized learning experience better suited for their capacity and instant feedback. School administrators can improve student performance, such as pass rate and proficiency. Clark, Kaw and Braga Gomes, (2022) advise adaptive learning give best results when it combined with pre class sessions.

Ennouamani & Mahani, (2018) have summarized adaptive learning systems to 3 models. They are Learning model, Adaptation model and Domain model. The learner model contains the student characteristics such as learning style, reasoning style, interests and student performance history. The domain model contains knowledge of the studying domain, study materials and learning objectives. The adaptation model has adaptation rules that align with the student performance and domain. It asses the student behavior and navigates them to relevant materials in the domain model. A sophisticated adaptive learning system temporally updates its rules and gets feedback from external and internal learning environments. As shown in the Figure 1.1 adaption model feeded by the leaner model and domain model. Then it provide adaptive feedback to the system. System interact with the leaner via graphical user interface. Adaptive model could suggest learner to attempt a easy or hard question, spend more time on basics or take a brake and start learning later.

Adaptive learning positively impacts student performance with empirical evidence, but it depends on the design of the adaptive learning system (M. Liu et al., 2017). It should be user-centric, and content must properly align with the learning outcomes. An adaptive learning system should be able to provide meaningful feedback and navigate students only to the relevant content.

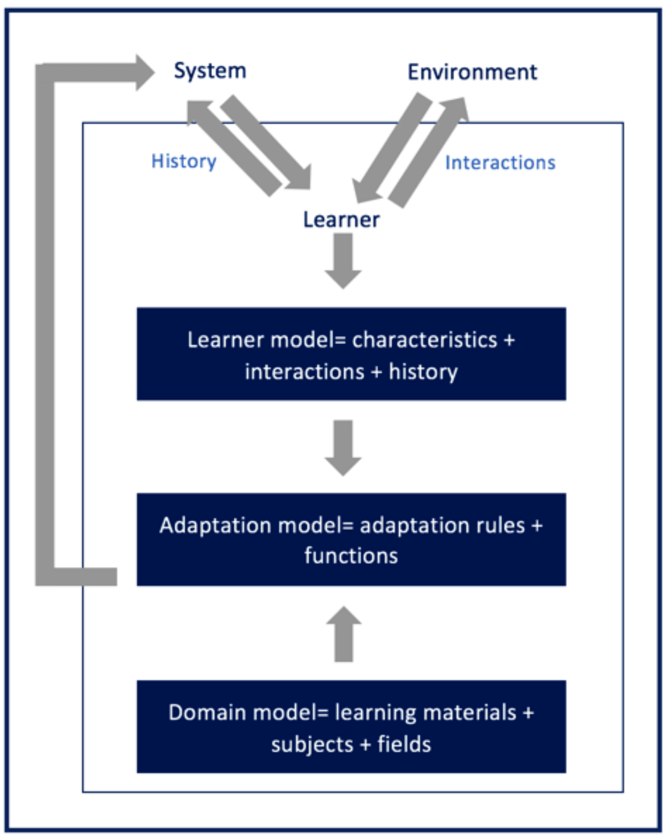

Figure 1.1 Adaptive e-learning systems' components (Ennouamani & Mahani, 2018)

According to (Martin et al., 2020), when educational institutes adopt adaptive learning methods, they face three challenges with respect to technology, instruction, and management. There are technological barriers when schools connect existing learning management systems to adaptive learning methods, real-time data-sharing challenges, and the complexity of adaptive systems. Teachers and instructors not having enough experience can lead to the adaptation of adaptive learning methods. Educational institutions must train and monitor how well they adopt adaptive learning methods. Sometimes educators resist adopting adaptive learning methods due to differences in the curriculums, additional workload, or not having confidence that adaptive learning methods can improve students' knowledge state. Lack of management support can also lead to adaptive learning method adoption failure. Incompatible organization goals or lack of leadership and insufficient human resources and financial resources can also cause to halt the implementation of adaptive learning systems.

1.3 Research problem

This research studies data sets from a real-world commercial adaptive learning platform. It provides practice questions and assignments targeting science and mathematics school curricula. Practice questions are called Goals on this platform. Each goal consists of multiple answer questions related to learning objectives. If a student gives the correct answer student will be allowed to proceed to the next question. If the student fails the question, he or she will get a new question or be presented with the study materials to refresh their knowledge.

This platform measures the mastery of a student using a modified version of Item Response Theory (IRT) (F. M. Lord, M. R. Novick and Allan Birnbaum, 1968), which is a statistical technique. This method consider only the questions difficulty ,student proficiency and skill discrimination ability of the question. Students ability to answer a question correctly depends on stundets mastery level on the skill represent by the question. But most of the skill have prerequisite skills. Exiting model does not consider the mastery level of prereuqisit skills. Subjected adaptive learning platform has not assessed the impact of study materials. Existing model does not consider the impact of study materials towards students performance. Hence there is requirement to explore novel method to measure students mastery level considering prequisits skill and study materials impact.

Current system provide lots of value informations to teachers such as mastery level achived by the students and the degree of effort each student have to put to reach the mastery level. This helps teachers to undestand individual students learning rate. If students are clustered based on the learning rate, teachers can analyze the class separate clusters and identify common poor skills among student clusters. This will help teachers rather than spend time on individual stundets weak areas, spend time one multiple students who has common weak skills.

1.4 Research gap

In literature, knowledge tracing is widely researched under many branches. In the early stages, Bayesian knowledge tracing (KT) was the most popular method for knowledge tracing method. Later IRT was introduced, and recently with the boom of deep learning, deep knowledge tracing was introduced. DKT outperformed all previous techniques, and there are many applications under all the branches. They predict students' ability to answer a question correctly, recommend learning materials or questions, assess the quality of the education, and many more.

When our data set is compared to the literature, our data set also has the sequence of questions under different learning objectives and the correctness of the answers like in other studies. One specialty in our data set is, middle of the question sequence, students referred to learning materials if they have poorly performed for the related learning objective , and attempted again. In the previous research work study materials are not included in the research problem. This can be used to measure the quality of the learning materials and how it impacts each student. Additionally, we attempt to incorporate question difficulty into the problem formulation.

In terms of learner characteristics, this research analyzes the possibility of clustering the students based on their prior knowledge and performance. The proposed study will also analyze the impact of study or the instruction materials provided to shape the leaners characteristics.

Table 1.2 Research Gap

Research Gap	Existing studies	Contribution of this study
Comparison of model perfoamce with and without question embeddings.	(Y. Liu et al., 2020) and (Song et al., 2022) have used quesrion embedding and learning object embeddings to improve the model performance. But they have not shown the improvement made by adding the question embeddings.	Other studies that utilized questions did not compare model performance without their inclusion. We have examined how model performance changes when question-level data is included versus when it is not.
Use of immediate student interactions instead of most related interactions.	(Y. Liu et al., 2020) has incorporate question embedding to improve model performance. But they have used similarity between the question embedding to predict target question answer correctness. This increase the computational requirement as it has calculate the similarity between questions and index them.	Other studies have considered students' performance of similar questions/learning objectives to the target question. But we have implemented novel approach by considering only the most recent stundet interactions to the target question. It reduce the model complexity and the computational requirements.
Prior knowledge of questions not required (other than relation to the knowledge graph)	(Song et al., 2022) have used question emeddings along with other information related question such as difficulty of the question, time taken to answer the question etc.	Proposed method only need to know the respective learning objective of the question.

1.5 Research question

1. What factors influence students' personalized learning experience within an adaptive learning environment?

2. How does choice of learning materials affect students' personalized learning experience in an adaptive learning environment?

1.6 Research objectives

1. Evaluate the effectiveness of utilization of question and learning objective relationship towards improving student mastery level.
2. Explore the potential of deep learning techniques in enhancing personalized learning experiences for students.

1.7 Research scope

The scope of the study is to analyzes a real-world dataset from an adaptive learning platform to predict students' mastery level. This study uses graph neural network to produce embeddings to represent questions and learning objectives. Such embeddings used to model student interactions as a sequence of questions and answers. This research aim to produce a model that can predict students ability predict answer next question correctly based on the previous sequence of questions and answers. We also compare the impact of using question embeddings to the model performance.

2 CHAPER 2 - LITERATURE REVIEW

2.1 Adoptive learning

“ Adaptive learning as an educational technology is a kind of scaffolding technique customized to help all stakeholders in an educational institution, teachers, students and school administrators” (Castañeda & Selwyn, 2018)

According to Ennouamani and Mahani, (2018) there are 3 main adaptive learning approaches. They are ;

- Macro-Adaptive Approach - This approach allows the user to move between courses at an adapted rate. It also considers the learning objectives and cognitive and intellectual characteristics. The instructor has to initiate the narrative.
- Aptitude-Treatment Interaction (ATI) Approach - This approach identifies the learner's aptitude and then alters the course of action to improve the learner's abilities. These systems can be used to develop Intelligent Tutoring Systems by generating learning materials suited to individual learner's capabilities.
- Micro-Adaptive Approach - This approach analyzes the learner and understands the learner's requirement or knowledge gap. It is a more dynamic system that considers real-time characteristics of the learners.

This study focuses on building Aptitude-Treatment Interaction (ATI) Approach using deep learning. The ATI approach emphasizes the user's control over the learning process. Studies have shown that the success of self-control in learning depends on the learner's abilities, suggesting that it may be beneficial to limit control for students with low prior knowledge and enhance it for high-performing students. It introduces three levels of control: complete independence, partial control within task scenarios, and fixed tasks with controlled pace.

Intelligent Tutoring Systems (ITS) utilize the ATI approach to detect users' skills. ITS implementation is based on adaptive e-learning system architecture, comprising the learner model and domain model. An adaptation model is used to generate and present adapted materials to each learner. This approach is also applied in adaptive hypermedia systems, where the goal is to design learning solutions that integrate hypermedia content in ITS to tailor it to individual learner profiles

2.2 Knowledge tracing

Human teachers can measure students' level of understanding and take necessary actions to fill the gaps. In the computer base teaching era, machines must learn the students' degree of understanding and take action to fill the knowledge gap. Abdelrahman, Wang, and Nunes (2023) Recognize this process as **Knowledge Tracing (KT)** . These KTs are widely used in Massive Open Online Courses (MOOCs), Intelligent Tutoring Systems (ITS), educational games, and adaptive learning platforms. However, capturing student knowledge level is not easy because questions can require multiple skills, dependency among skills, and forgetting or decaying knowledge over time. Since John R. Anderson introduced knowledge tracing in 1986, researchers have attempted to develop many machine-learning models to solve KT.

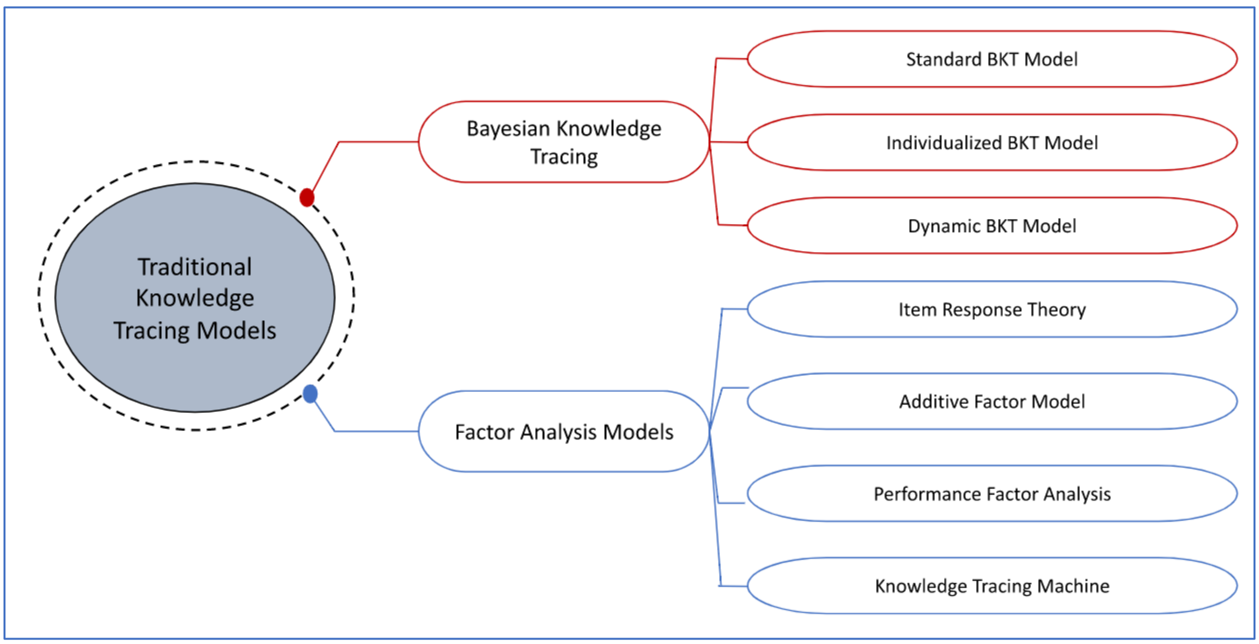

Figure 2.1 Traditional Knowledge Tracing Models

2.3 Bayesian Knowledge Tracing

First generation of Traditional KT models were based on Bayesian Knowledge Tracing (BKT). Bayesian Knowledge Tracing (BKT) is inspired by mastery learning, which assumes that all students can achieve mastery of a skill under two conditions:

- Knowledge is organized as a hierarchy of skills, and

- Learning experiences are structured to ensure mastery of lower-level skills before moving to higher-level ones.

BKT models typically employ probabilistic graphical models like Hidden Markov Model and Bayesian Belief Network to track students' evolving knowledge states as they practice skills.

First BKT model was developed by Corbett and Anderson in 1994 (Albert T. Corbertt & John R Anderson, 1994). It considered two statues of student learned or unlearned. This model assumed that students do not forget what they mastered. . But it considers the probability that students may guess the answer p(G) or mistakenly select the wrong answer (slip) p(S).

This model consider as the standard BKT model. It has four parameters.

Table 2.1 Bayesian knowledge tracing model parameters

Parameter	Description
p(L0)	Probability of skill mastery by a student before learning
p(T)	Probability of transition from an unlearned state to a learned state
p(S)	Probability of slipping by a student in a learned state
p(G)	Probability of guessing correctly by a student in an unlearned state

At each time step $n \geq 1$, the model estimates the probability p(Ln) of skill mastery by a student by

Equation 2.1 Bayesian Knowledge Tracing model

$$p(L_n) = Posterior(L_{n-1}) + (1 - Posterior(L_{n-1})) * p(T)$$

$$Posterior(L_{n-1}) = \begin{cases} \dfrac{p(L_{n-1}) * (1 - p(S))}{p(L_{n-1}) * (1 - p(S)) + (1 - p(L_{n-1})) * p(G)} & \text{if the } n\text{-th attempt is } \textit{correct}; \\ \dfrac{p(L_{n-1}) * p(S)}{p(L_{n-1}) * p(S) + (1 - p(L_{n-1})) * (1 - p(G))} & \text{otherwise.} \end{cases}$$

2.4 Factor analysis models

Factor analysis models are the second branch of traditional knowledge tracing methods. It plays a vital role in measuring assessments. Factor analysis models are based on Item Response Theory

(IRT). This study use a data set from a commercial adaptive learning system that measures students proficiency using IRT. Our data set assed using a modified version of IRT which has a memory of previous performance and it helps to reach the mastery level based on the students adaptive rate.

Item Response Theory is a psychometrics method, which means it is statistical framework to analyze and understand the properties of individual test items/questions and the performance of test-takers on each item. It is introduced by F. M. Lord et al., 1968. IRT performance as a logistic function.

According to IRT every question has a degree of difficulty and student has a level of ability. Below equation is the basic form of ITR. p_{ij} is the probability of student *i* answering correctly to the question *j*. a_i is the ability of student *i* and b_j is the difficulty of the question *j*.

Equation 2.2 Item Response Theory formula

$$p_{ij} = \frac{e^{a_i - b_j}}{1 + e^{a_i - b_j}}$$

Assumptions in IRT;

- Probability of student correctly answering a question model as an item response function
- Item response function monotonically increase with respect to the ability of the student
- Questions are conditionally independent.

2.5 Deep knowledge tracing and Graph neural network

Piech et al., (2015) Lead the **Deep Knowledge Tracing (DKT)** introducing deep knowledge tracing **.** DKT mainly uses deep learning to predict students' ability to answer a question correctly. DKT models have outperformed traditional knowledge tracing models. From the machine learning perspective knowledge tracing is sequence modeling task. It try to predict the next state of the sequence (students ability to answer the next question given the previous questions and performance). Hence deep learning models have use Recurrent Neural Networks (RNN) or Long Short Memory (LSTM) to model the sequence. There are many branches under DKT.

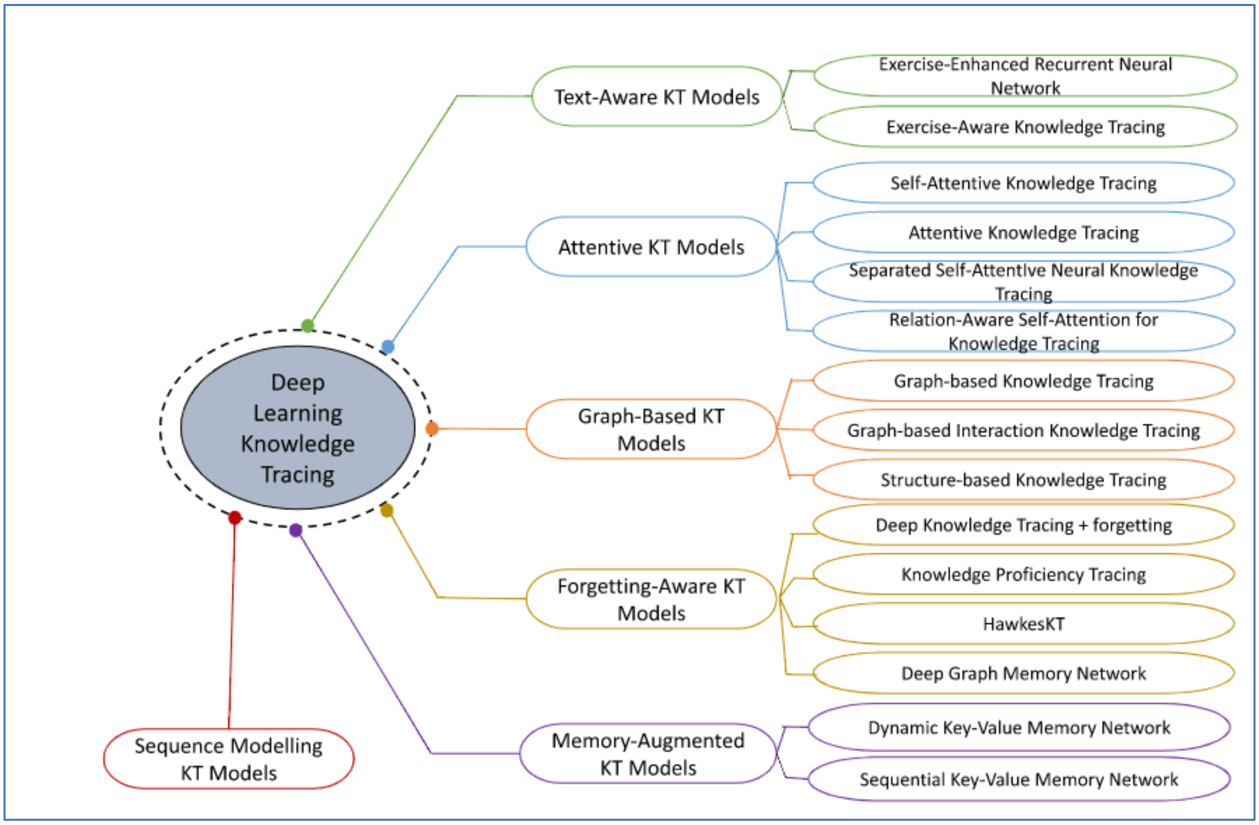

Figure 2.2 2 Deep Learning Knowledge Tracing models

Despite its encouraging performance DKT models had some drawbacks. They cannot predict the outcome of questions related to multiple skills. Also, it can not model the connection between multiple skills. It also assumes that all the questions are related to each other with the same probability which is not likely to happen all the time. Researchers try to overcome these limitations using Extended-deep knowledge tracing, which introduces by (Piech et al., 2015). They added additional student features such as previous knowledge, question answering rates and time spent on learning and practice; and, exercise features, such as textual information, question difficulty, skill hierarchies and skill dependencies.

But these limitations successfully overcome by Graph based Knowledge Tracing models. They can integrate the relationship between knowledge concepts and questions. Nakagawa et al., 2019 introduced Graph based knowledge tracing. They present knowledge concepts by nodes and relationships between them using edges. They formulated the problems as time series classification problem at node level.

According to (Abdelrahman et al., 2023) there are three main graph-based KT models. They are

- graph-based knowledge tracing
- graph-based interaction knowledge tracing
- structure-based knowledge tracing (SBKT)

This research leans toward structure-based knowledge tracing as we use knowledge graphs representing relationships between knowledge concepts (KC/learning objective (LO) as per our data set).

Tong et al.,(2020) introduced the structure-based knowledge tracing method. They have tried to solve two main challenges in this paper. They are the temporal impact of exercise sequence and the spatial impact of the knowledge structure or knowledge graph. In order to solve these challenges, they have introduced structure-based knowledge tracing(SBKT). SBKT can simultaneously model the temporal and spatial impacts.

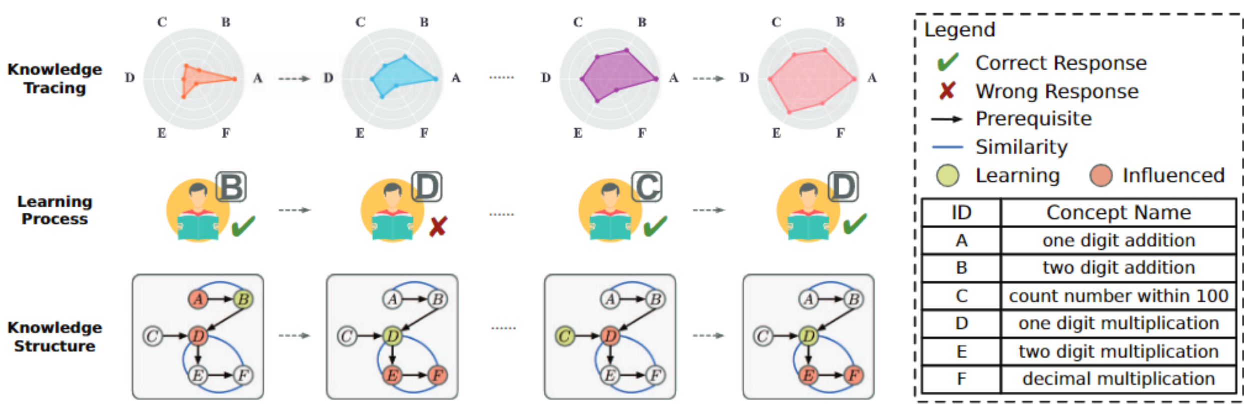

Figure 2.3 structure-based knowledge tracing (Tong et al., 2020)

Figure 2.3 depict sequence of exercises related to one knowledge structure. Under this structurer there are connected concepts. They are either prerequisites or similar concepts. As the student proceed with the question students knowledge statues of each concept change. It is shown in the radar map in the top. Changes in radar map shows the temporal impact of the students' knowledge statues and knowledge structures shows how responses impact the learning concept and related(influenced) concepts, which is the spatial impact.

2.5.1 Graph Neural Network

The rapid development of internet technology and web applications has led to a vast amount of data being generated on the internet, which can be used to create valuable knowledge. Such knowledge lead to create knowledge graphs. Graph Neural Network (GNN) created to learn from such knowledge graphs and predict the unknown. GNN are a class of deep learning methods designed to perform inference on data described by graphs. They are neural networks that can be directly applied to graphs, allowing for node-level, edge-level, and graph-level prediction tasks (Ye et al., 2022).

The message parsing process is what allows GNNs to learn from the structure of the graph. By sending messages to each other, the nodes in the graph are able to share information about their local neighborhoods. This information can then be used to update the nodes' states, which in turn can be used to make predictions about the graph. There are a variety of different message parsing functions that can be used in GNNs. The choice of message parsing function depends on the specific task that the GNN is being used for. For example, if the GNN is being used to predict the relationship between two entities, then the message parsing function might be designed to extract features from the entities and their relationships (Serra & Niepert, 2023).

2.6 Leaners characteristics

Hemmler and Ifenthaler, (2022) have identified internal and external indicators of the learning context for supporting adaptive learning. Based on the authors internal dimensions, Past performance is a one dimension that support toward adaptive learning. It can be measure through previous grades, rank, previous experience with the course content, prior credits and course repetition. All these indicators are included in our data set. Additionally under skills and abilities dimension, prior knowledge indicator also captured in our data set. In contrary there are many other dimensions such as demographics, learning approach, emotions, perception towards teacher/course and etc. Hence our study limited only to student performance and skill/abilities dimension when analyzing learners characteristics in an adaptive learning environment.

Afini Normadhi et al.,(2019) summarize learners personal traits in 3 main domains and the relevant sub domains.

- Cognition – learning style /cognitive style/ prior knowledge/ personality type/thinking process/working memory capacity.
- Affective – emotions/ mental state/ engagement
- Behavior/psychomotor – cognitive abilities/ performance

Our study based on performance under Behavior/psychomotor and prior knowledge under cognition.

Authors conclude most of the adaptive learning environments build on personal traits under cognitive learning domain. Most frequently used personal trait identification method is computer based detection using machine learning (majority) , without machine learning or hybrid approach. Authors mentioned most of the research work suffer with small sample size which address in our study. And our work intend to use knowledge graph based approach which was not used mention in (Afini Normadhi et al., 2019) literature review from 2007-2017.

Hsu,(2012) developed Learning Effort Curve Mode using dynamic real-time based learning effort quantification technique (related work from the same author). This author has used learning style, learning efficiency and self-efficacy as learner characteristics. In the evaluation author has grouped 125 students in to 16 groups and measured Learning Effort Curve Mode. Author has found, despite the learning style or characteristics, descending learning effort leads to ascending learning performance for high learning efficacy groups . Similarly ascending learning effort leads descending learning performance low learning efficacy groups.

2.7 Recommendation system

Rule-based filtering systems rely on manually or automatically generated decision rules that are used to recommend items to users. Content-based filtering systems recommend items that are considered sufficiently similar to the content descriptions in the user profile. Collaborative filtering systems, also referred to as social filtering, match the rating of a current user for items with those of similar users in order to produce recommendations for items not yet rated or seen (Duval et al., 2007)

2.7.1 Study material recommendation

Duval, Klamma and Wolpers, (2007) developed an advance recommendation engine to recommend links to students in an E-learning platform. Regular recommendation engines, consider all the users logs at once to recommend links using sequential pattern mining algorithms. These authors have clustered users using k-means clustering algorithm (2-5 clusters) considering number of pages visited and the average knowledge obtained from these pages. Then they have applied AprioriAll, GSP and PrefixSpan sequential pattern mining algorithms for each cluster to generate recommendation rules. This new approach have generated similar or more rules for the same support and with high confidence compared to using all user data at once. As per the conclusions, GSP and PrefixSpan algorithms have shown better slightly better results when there are 2 or 3 clusters. In our approach we can generate 2 or 3 clusters to identify similar students. These authors haven't consider the learning objectives but students navigation through the web site. Our work can also consider the number of questions and instruction materials referred and the student progress in the learn path (similar to average knowledge) as features for the clustering algorithm. Our data set do not contain students activity log but students performance in relation to learning objectives. And the due graph nature of our data set make it more complex to analyze.

Borges and Stiubiener, (2014) developed a recommendation system to suggest learning materials to students based on the learning style of the students and the relevant learning objectives. Authors have clustered the students based on their learning style, they have identified 6 learning styles based on input , perception and process (Richard Felder, 2002), and how different learning materials associated with the learning style. Then utility function developed to measure the distance between learning objectives and learning style(LS) using Manhattan distance. Utility function range from 0 to 6, 0 indicate no difference between LO and LS. 6 indicate LO and LS is totally different from each other. Based on utility function results they and LS they suggest the learning materials. They have tested this system with 28 students and 362 recommendations, 89% of the students are satisfied with the results. In their research , they have not considered the students performance and applied for a small student group. Contrary in our study we consider students performance history and student performance after referring the learning materials. Our study based on large pool of students. Additionally we map LOs with knowledge graphs and how student performance related to each LO.

3 CHAPTER 3 - METHODOLOGY

3.1 Data

This research uses a real-world data set from an International E-learning (courseware) platform that uses state of the art adaptive learning technology. This platform provides educational content targeting schools for Mathematics, Economy, Chemistry, Biology, Physics and Psychology. Based on the research question, identified data was already collected with the organization’s approval.

Subjected Adaptive Learning Platform (ADP) measures the learners' progress level ranging from 0 to 100. Teachers can assign assignments to the student related to a specific Learning Objective(LO). A student has to reach 100 progress to complete the assignment, then the student has achieved the ‘Mastery’ to that LO. Each LO has minimum 4 question, progress of a student for a given LO is

Progress = proficiency score x fraction of the minimum questions learner have tried

If student fail master a LO, student get to do more practice questions. If the student need further support, he or she get more instructions and direct back to the prerequisite LOs.

All the learning objectives, concepts, questions, and course materials are associated to knowledge graphs. These knowledge graphs and progress levels drive the students journey to master a given learning objective. But other characteristics of the student joinery are not considered. Such as time spent on a question, time spent on instructions, quality of the instruction materials, etc.

Table 3.1 Summary of Data

Data	Number of data points	Attributes
Student coursework performance	3.3 million	• Learning objectives • coursework id • user id • progress • question id

		• correctness of the answer • time spent to answer • time spent for the question instruction • study material id referred
Student assignment	140,000	• Learning objectives • test id • user id • question id • correctness of the answer
Learning objective map (knowledge graph)	1145	• Source LO Id (prerequisite LO ID) • Destination LO Id • Source LO Title (prerequisite LO Name) • Destination LO Title

3.2 Solution design

3.2.1 Selection of solution architecture

According to the literature authors have used different methods to solve knowledge tracing. There are mainly two methods. First method is Traditional knowledge tracing which has two branches. They are;

1) Bayesian knowledge tracing
2) Factor analysis models

Second method is Deep knowledge tracing. This is the latest knowledge tracing methodology, and it has outperformed Traditional knowledge tracing methods. Our dataset has already tested with modified item response theory which is one of the models under Factor analysis models. Hence Traditional knowledge tracing methods will not be used for this research. Instead, Deep knowledge tracing methods will be employed expecting better performance.

Under Deep knowledge tracing there are multiple models. All these models use Deep Neural Networks with different input types and different neural network architecture. Subjected data set has heterogenous data types and relationship between these data better explained by Graphs/Networks. Hence this study will use Graph based knowledge tracing methodology to predict students' knowledge level. There are multiple graph based knowledge tracing methods in literature and this study will compare and contrast different model when building the model.

3.3 Graph Neural Network (GNN)

The evolution of Graph Neural Networks (GNNs) has given rise to numerous applications across diverse domains, including but not limited to Natural Language Processing (NLP) Computer Vision (CV), and Recommendation Systems . GNNs, with their capacity to capture high-order information, have paved the way for substantial advancements in these fields. In our research, we harness the power of Graph Convolutional Neural (GCN) within our Graph-based Interaction Knowledge Tracing (GIKT) model. By employing GCN, we aim to extract meaningful relations between skills and questions, effectively translating them into rich and informative representations. As far as our knowledge extends.

3.4 How GNN works

Graph Neural Networks (GNNs) are a type of deep learning model designed to work with graph-structured data, where data is organized as nodes connected by edges (like a social network, a road map, or a recommendation system). GNNs aim to understand and process this data effectively.

Each node in the graph starts with an initial representation, typically as a vector of numbers. These initial representations capture the characteristics of each node.

GNNs operate through a process of message passing. At each step, each node sends messages to its neighboring nodes. These messages typically contain information about the node itself and its immediate neighbors. The idea is that nodes can exchange information and learn from each other.

After receiving messages from their neighbors, nodes aggregate this information to update their own representation. This aggregation process combines the information from the node itself with that of its neighbors. This is done by Neural Networks. In the below Figure 3.1 gray boxes show the neural networks.

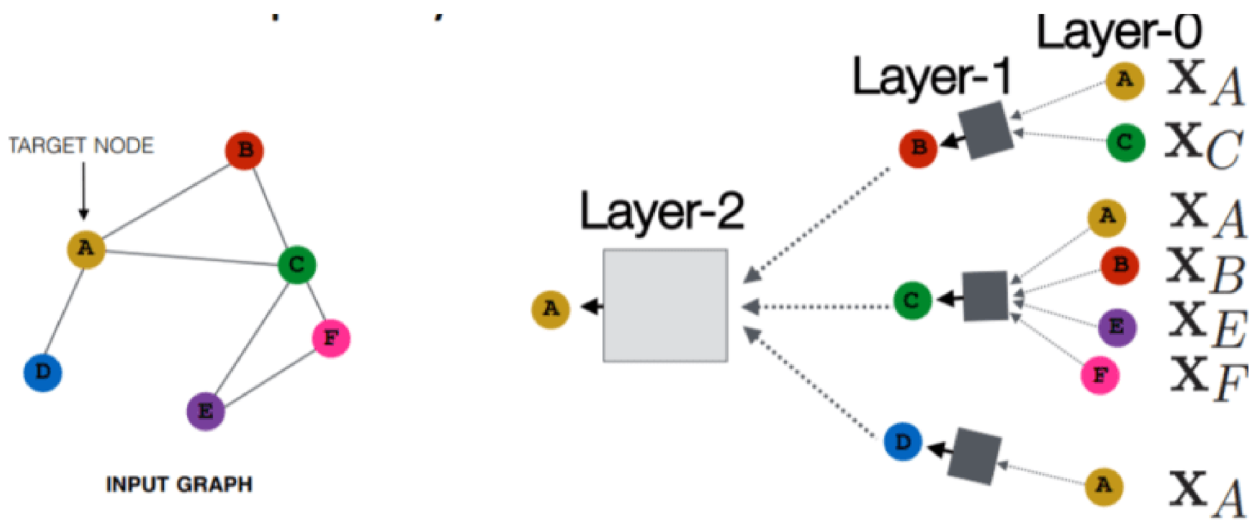

Figure 3.1 - Graph Neural Network (source - Stanford Graph based Machine Learning lecture slides)

For example, (X_A) is a feature vector of node A. The inputs are those feature vectors, and the box will take the two feature vectors (X_A and X_c), aggregate them, and then pass on to the next layer.

$$h_v^0 = X_v \ (feature\ vector)$$

Equation 3.1 - feature vector

Notice that, for example, the input at node C are the features of node C, but the representation of node C in layer 1 will be a hidden, latent representation of the node, and in layer 2 it'll be another latent representation. At each kth layer, h_v^k feature vector produced by the Equation 3.2. It average the previous layer by the number of nodes in the current layer and add bias to previous layer, then perform a nonlinear activation denoted by σ. W_k (weight matrix) and B_k (bias matrix) are trainable parameters.

$$h_v^k = \sigma(W_k \sum \frac{h_u^{k-1}}{\mid N(v)} + B_k h_v^{k-1}) \text{ where } k = 1, ..., k-1$$

Equation 3.2 - Neighborhood aggregation

3.5 Data preparation

3.5.1 Creating knowledge graph

Knowledge graph contain homogeneous nodes representing both questions and learning objectives. Learning objectives have proceeding or prerequisites learning objectives. Hence learning objective nodes has directional edges starting from prerequisites nodes. Every question has one learning objective. One learning objects have multiple questions. All the edges between questions and learning objectives, start from a learning objective.

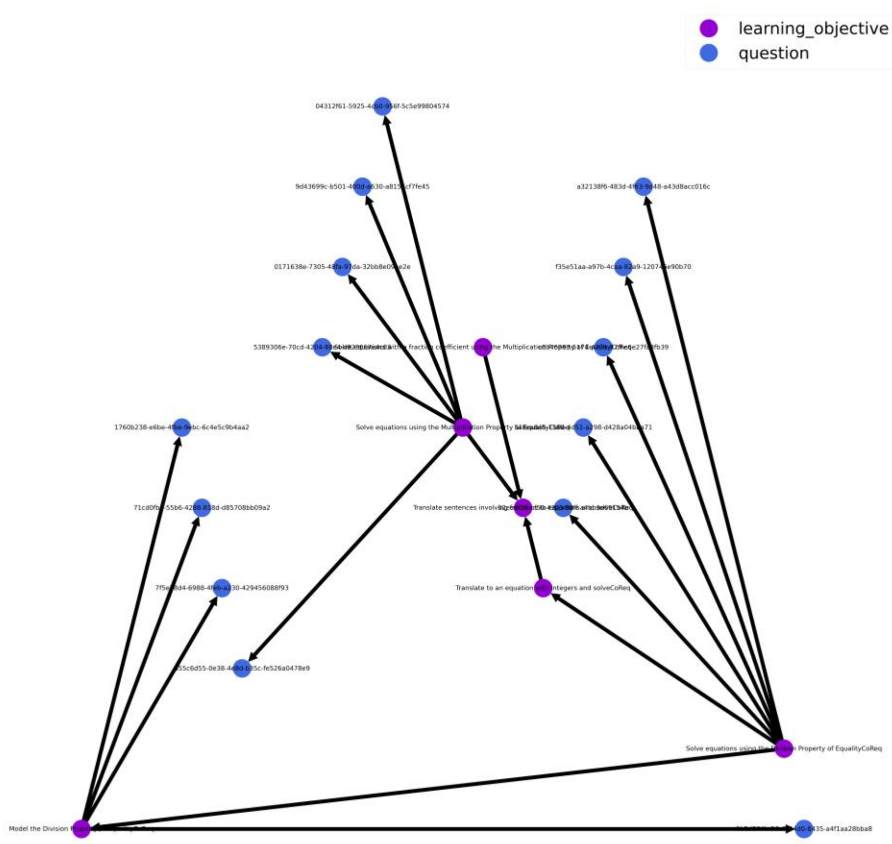

Figure 3.2 Sample Knowledge Graph

3.5.2 prepare students questions sequence.

In the context of our study, the data sets and learning objectives is centered within the domain of Mathematics. This domain encompasses various sub-domains, including but not limited to Calculus, Trigonometry, and Complex Numbers. Within each sub-domain, multiple learning objectives exist, each comprising a set of questions with varying degrees of difficulty.

Instructors can curate assignments by selecting specific learning objectives from a chosen sub-domain. The existing system, in turn, leverages student performance metrics to dynamically select questions from the question bank. For the purpose of our study, an assignment is chosen, and the students are divided into training and testing subsets. 80% of students are randomly assigned to the training group, while the remaining 20% constitute the test group.

Each student's interactions with the system are considered as sequences, with the defining features being the learning objective identifier and the question identifier. The target variable for analysis is the correctness of the answers provided by students. Notably, students engage with varying numbers of questions, resulting in non-uniform sequence lengths. Given that Long Short-Term Memory (LSTM) layers, integral to our methodology, exhibit suboptimal performance with excessively long sequences, we adopt a strategy to mitigate this challenge.

To address the issue of sequence length, each student interaction sequence is decomposed into multiple smaller sequences. To ensure the preservation of these reduced sequence lengths, padding is applied. Throughout the training and testing processes, sequences padded with additional elements are appropriately masked to prevent their undue influence on model performance. This approach maintains the integrity of the data while accommodating the architectural considerations of the LSTM layer.

3.6 Proposed method

In our proposed methodology, we employ a systematic approach to enhance the understanding and prediction of student performance. The key steps of our method are outlined as follows:

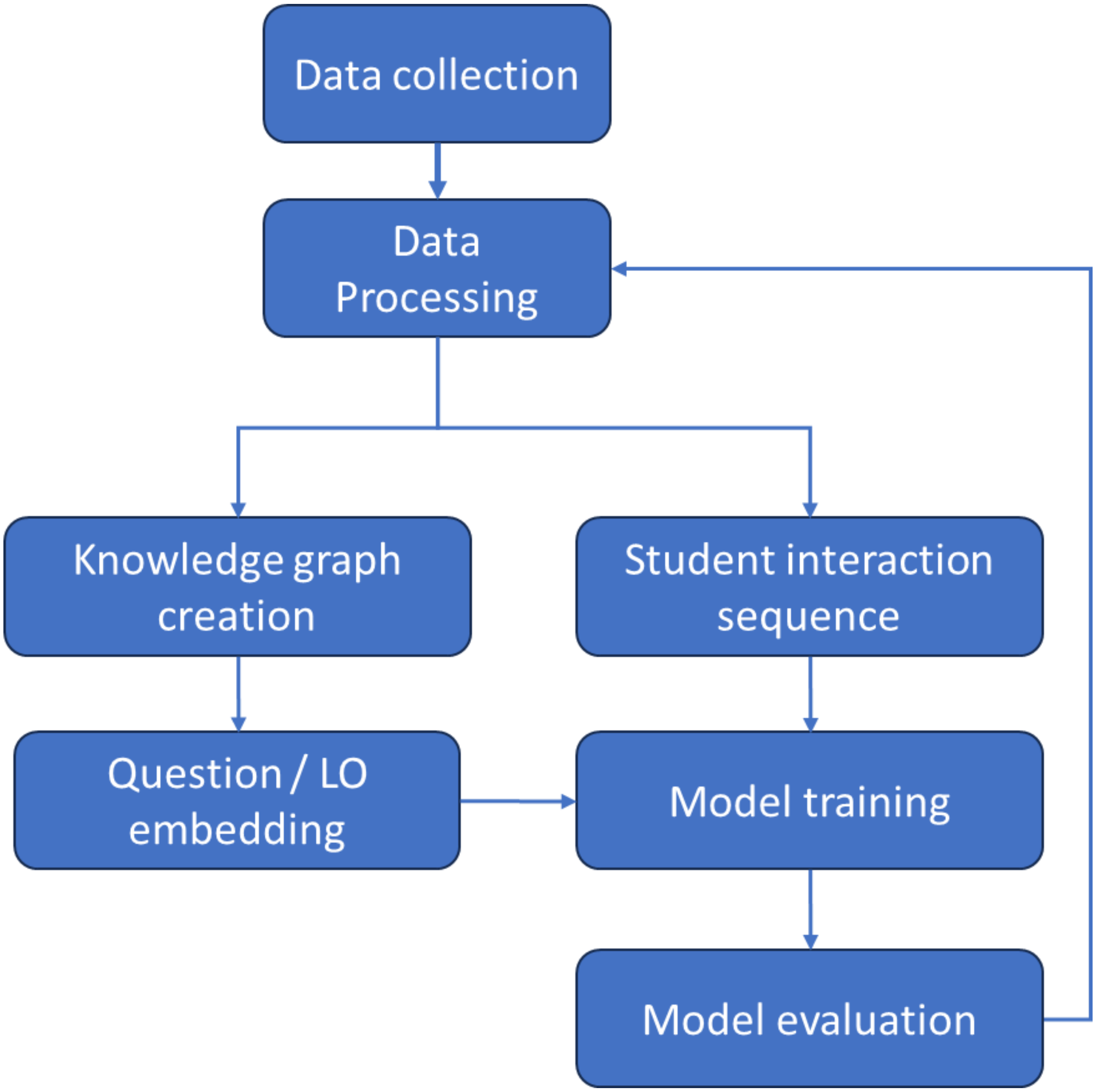

Figure 3.3 Proposed Methodology

1. Knowledge Graph Creation:

We initiate the process by constructing a knowledge graph. This graph serves as a structured representation of the relationships between various entities in the Mathematics domain, encompassing sub-domains such as Calculus, Trigonometry, and Complex Numbers. Nodes within the graph represent learning objectives and questions, while edges signify the associations between them.

2. Embedding Layer Generation:

To facilitate the learning process, we generate an embedding layer for both questions and learning objectives. This embedding layer translates the discrete entities into continuous vector representations, enabling the model to capture nuanced relationships and semantic meanings within the knowledge graph.

3. Data Set Splitting:

The available data sets are divided into distinct training and testing sets. Approximately 80% of students are allocated for training purposes, ensuring the model is exposed to a diverse range of interactions. The remaining 20% constitutes the test set, providing an independent evaluation of the model's predictive capabilities.

4. Node Embedding Retrieval:

In the model training phase, we retrieve embeddings for relevant nodes associated with the current question (t) and its preceding questions (t-1, t-2, ..., t-n). This retrieval process is crucial for capturing the dependencies between questions and learning objectives.

5. LSTM-Based Prediction:

Leveraging Long Short-Term Memory (LSTM) layers, our model is equipped to predict the correctness of students' answers. The LSTM architecture proves advantageous in capturing sequential dependencies within the student interaction sequences. The model processes the embeddings of relevant nodes over time, allowing it to discern patterns and make informed predictions regarding the correctness of students' responses.

3.7 Model Architecture

Model accepts question id of the t^{th} question (Q_t). Then the knowledge graph return the relevant Knowledge Graph Embeddings (KGE) to the Long Short Memory layer (LSTM). LSTM layer produce 2 outputs. One is hidden layer which pass through the sequence. Other output is the mastery level which goes through a binary classifier that convert mastery level to answer correctness using a sigmoid function. Predicted answer correctness (P_t) and actual value of answer correctness (C_t) at t^{th} time are used to calculate the Loss value using predetermined loss function. These loss values used to optimize the model parameters using an optimizer.

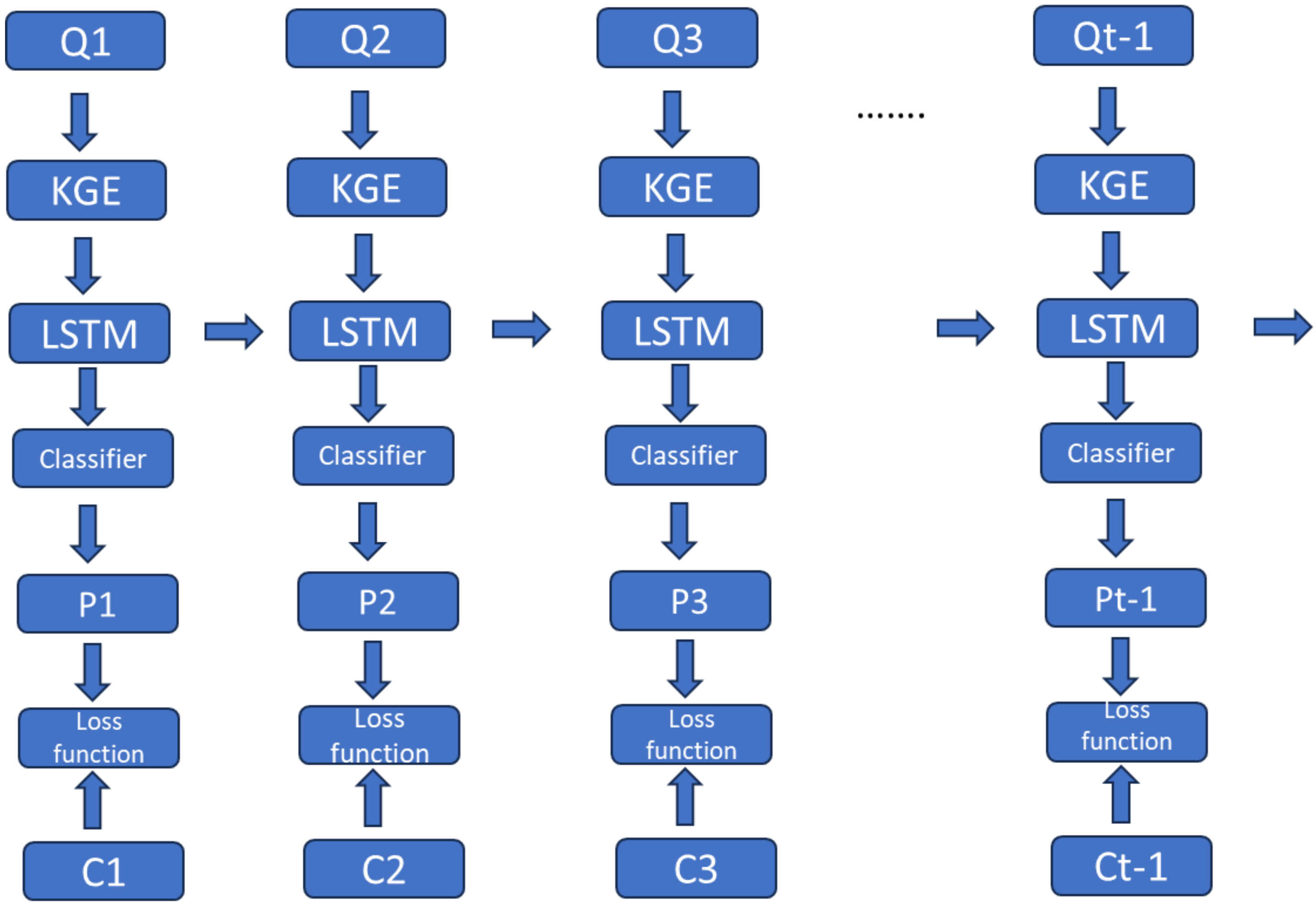

Figure 3.4 Model Architecture

4 Evaluation and Discussion

As discussed in early chapters preliminary task of this research is to predict student ability give the correct answer to a question given a sequence of previous questions, correctness of answers and related learning objectives. For given question there are two outputs. They are correctly answered or not. Hence this is a binary classification problem. We develop mainly two models and use one benchmark model (BKT). It is equally important to predict students being able to give answers correctly or wrong. And dataset is approximately balance data set.

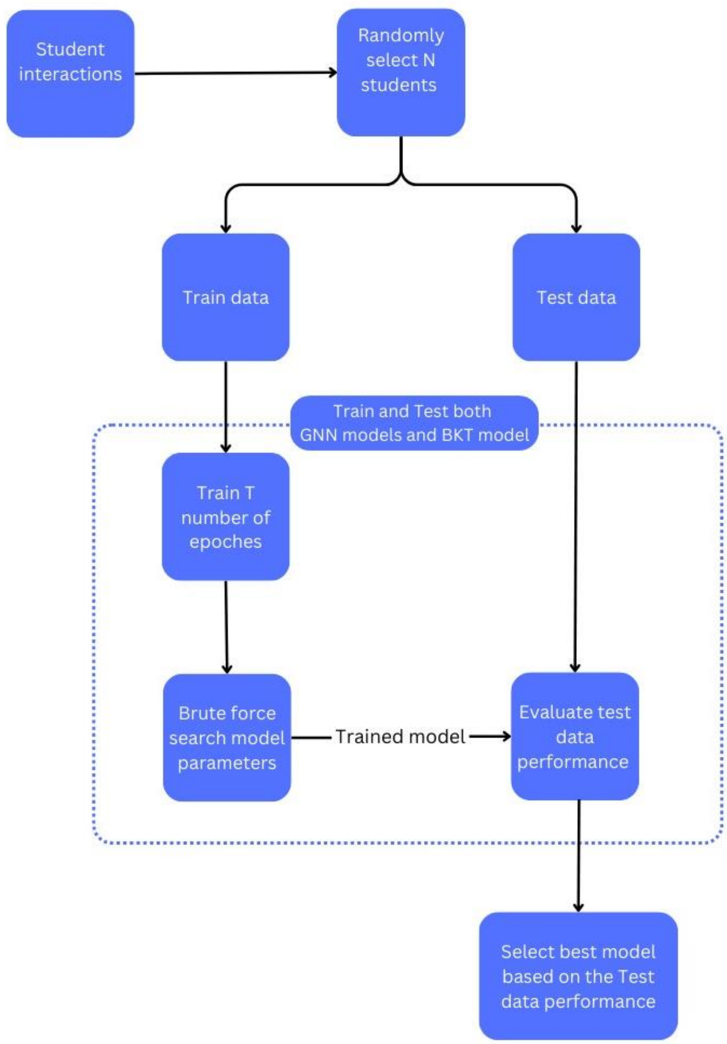

Figure 4.1 Evaluation process

4.1 Accuracy

Accuracy is a commonly used metric for evaluating the performance of binary classifiers. It is defined as the proportion of correct predictions made by the classifier. In the context of a binary classifier, accuracy is calculated as shown in the Equation 4.1

$$\text{Accuracy} = \frac{\text{Number of correct positive predictions } + \text{ Number of correct negative predictions}}{\text{Total number of predictions}}$$

Equation 4.1 Accuracy

Accuracy is a good metric to use when both true positives and true negatives are equally important because it considers both types of correct predictions. Therefore, we use accuracy to evaluate individual model performance. It also frequently used in related literature to compare models.

4.2 Area Under Receiver Operating Characteristic Curve

We also use Area Under Receiver Operating Characteristic Curve (ROC AUC) to compare models and select best model parameters. Receiver Operating Characteristic curve plots true positive rate vs false positive rate. ROC AUC, or Area Under the ROC Curve, is a performance metric for binary classification problems. It measures the ability of a classifier to distinguish between classes and is used as a summary of the ROC curve. An ROC curve is a plot of the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. The TPR is the proportion of positive cases that are correctly identified as positive, while the FPR is the proportion of negative cases that are incorrectly identified as positive.

The AUC is calculated by measuring the area underneath the ROC curve. A perfect classifier would have an AUC of 1, meaning that it can correctly identify all positive cases and correctly reject all negative cases. A random classifier would have an AUC of 0.5, meaning that it is no better than guessing.

AUC is a useful metric because it is not affected by the class imbalance in the data. This means that it can be used to compare the performance of classifiers even when the number of positive and negative cases is not equal. Hence we use ROC AUC as the second performance indicate the compare models.

4.3 Loss function

The choice of a loss function is important in deep learning projects as it serves as a guidepost for the neural network to minimize errors during training. Essentially, it quantifies the disparity between predicted and actual values, allowing the model to adjust its parameters iteratively for improved performance. In binary classification tasks, where the outcome belongs to one of two classes, binary cross-entropy is a popular choice for the loss function. It calculates the difference between the predicted probability distribution and the actual distribution of the binary outcomes. By penalizing large deviations between predicted and actual probabilities, binary cross-entropy incentivizes the model to converge towards accurate classifications, making it a fundamental tool in optimizing the performance of binary classification models in deep learning projects. Our target variable has only values. Students gave the correct answer or not. Hence it is a binary variable. Therefore, we chose binary cross-entropy as our loss function. Loss function plotted over number of epochs (number of training loops) define as learning curve.

Loss function provide multiple insights about the model behavior such overfitting and underfitting and hyper parameter tunning, model convergence. Too many training epochs lead to overfitting the model. And fewer training epochs lead to underfitting the data. To find balance number of training epochs we use training loss and testing loss. We stop training when testing loss function stop reducing parallelly to training loss and start to increase. After examining several learning curves across varying epochs, we noticed a consistent pattern: the testing learning curve began to rise in comparison to the training learning curve after 10 epochs. As a result, we concluded that it was prudent to cease training beyond this point.

4.4 Optimizer

The choice of optimizer holds immense significance in deep learning projects, as it determines the efficiency and effectiveness of the model's training process. Optimizer, optimize weights and biases in a deep neural network based on the loss function values. Adam optimizer is one of the frequently used optimizers in adaptive learning domaining. After trying out different optimizers including Adam optimizer, we observed Adam optimizer tends converge model faster.

Among various optimizers, the Adam optimizer stands out for its adaptive learning rate capabilities and momentum-based adjustments, which often lead to faster convergence and improved performance. However, despite its advantages, the Adam optimizer has notable drawbacks. One significant concern is its sensitivity to hyperparameters, such as the learning rate, beta1, and beta2 parameters, which may require careful tuning to prevent performance degradation. Additionally, Adam may exhibit suboptimal performance on certain types of datasets or architectures, necessitating thorough experimentation and comparison with alternative optimizers to ensure optimal results in deep learning projects.

4.5 Split train and test data

When models are trained, models can be overfitted to the data. Hence, we split data in to test data and train data. Each student's interaction sequence has an order, similar to a time series. Therefore we select the first 80% of the interactions as the training data of each student. Rest of the 20% of the interactions of each student consider as the test data. Each model trained on train on train data and calculate the model performance. Then we predict the student answer correctness using test and calculate the model performance to observe whether model is overfitted.

During the training process we consider the running average loss of each epoch to select best model parameters. We use Binary Cross Entropy function as the loss function, since this is binary classification task. Considering the lengthy time each model takes to train we can not use grid search to find the optimal model parameters, but we use brute force method to find better model parameters. As an example we adjust model learning rate , hidden layer size , number of layers etc.

Since we are using a new data set we can not compare our model performance directly with the previous studies. Hence we use BKT and train and test the same data set. All most all the other studies have used BKT as a benchmark model. This can be used to compare out model performance and data set with other studies.

4.6 Experiments and Results

Two sets of experiments designed for this study. One set of experiments consider only learning objectives predict correctness of answers. Second set of experiments consider both learning objectives and questions. Both experiments conducted multiple times and calculate mean and variance of Accuracy and ROC_AUC.

4.6.1 Model Performance without Question Embeddings

In this experiment, model consider only Learning Objective Embeddings and students answer correctness. Model use embeddings of learning objective relevant to question in the question sequence and each questions answer correctness. Model does not significantly improve after 3rd epoch (test learning curve does not drop significantly after 3rd epoch). This model best performance is Accuracy 74.2% and ROC AUC 68%.

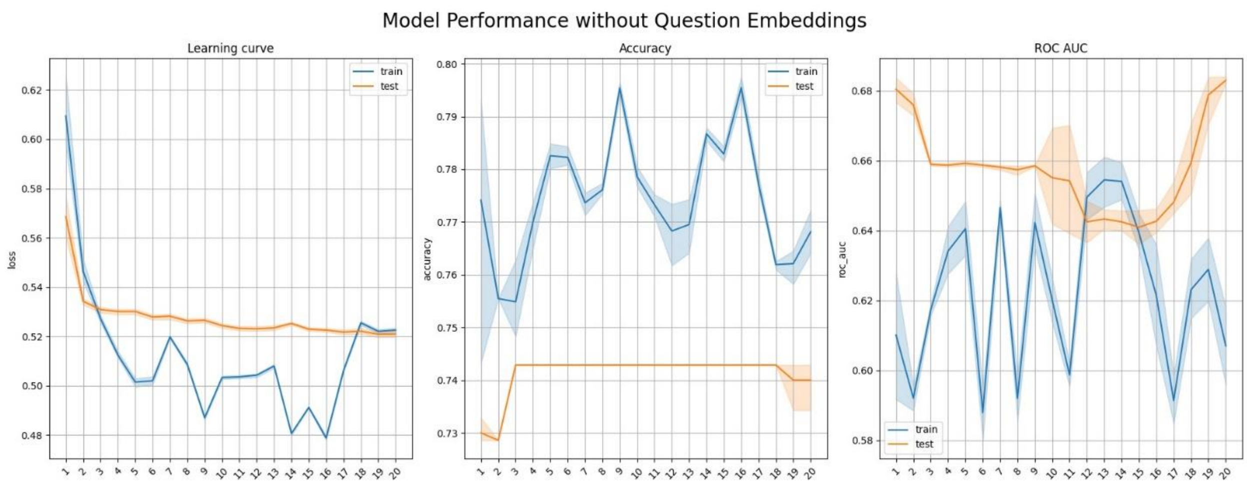

Figure 4.2 Model performance without question embeddings

	accuracy_test		accuracy_train		losses_test		losses_train		roc_auc_test		roc_auc_train	
	mean	std	mean	std	mean	std	mean	std	mean	std	mean	std
epoches												
1	73.714	0.782	77.386	1.399	0.539	0.015	0.538	0.041	66.398	0.737	62.092	2.032
2	74.286	0.000	78.299	0.903	0.528	0.002	0.504	0.012	65.771	0.194	61.789	2.356
3	74.286	0.000	77.690	0.795	0.525	0.001	0.497	0.011	64.229	0.463	64.115	2.708
4	74.286	0.000	77.219	1.254	0.523	0.001	0.511	0.019	66.099	1.988	61.117	2.046
5	73.714	0.782	76.696	1.314	0.534	0.011	0.534	0.037	66.881	1.266	62.031	2.351
6	74.286	0.000	78.107	0.840	0.525	0.001	0.504	0.013	66.287	0.963	61.802	3.331
7	74.286	0.000	77.657	0.644	0.522	0.001	0.498	0.012	64.796	1.352	63.507	2.417
8	74.286	0.000	77.359	1.362	0.520	0.001	0.511	0.020	66.182	2.062	61.085	2.283
9	73.714	0.782	77.046	1.988	0.539	0.015	0.537	0.037	66.589	0.965	62.510	2.003
10	74.286	0.000	78.155	0.855	0.527	0.002	0.505	0.011	65.591	0.469	61.537	2.736
11	74.286	0.000	77.595	1.172	0.524	0.001	0.497	0.011	64.094	0.545	64.078	2.279
12	73.714	0.782	77.317	1.596	0.523	0.000	0.510	0.019	67.159	1.663	61.940	1.818
13	73.714	0.782	76.950	1.700	0.540	0.018	0.541	0.043	66.654	1.126	61.609	2.021
14	74.286	0.000	77.993	0.749	0.527	0.002	0.504	0.011	65.250	1.201	61.382	3.044
15	74.286	0.000	77.658	0.934	0.524	0.001	0.498	0.011	64.002	0.482	63.622	2.417
16	74.286	0.000	77.121	1.599	0.521	0.001	0.511	0.019	66.225	1.891	61.068	2.356
17	73.857	0.639	75.631	2.671	0.542	0.023	0.547	0.054	66.788	1.242	61.175	2.726
18	74.286	0.000	78.045	1.017	0.527	0.002	0.504	0.012	65.882	0.021	62.380	2.569
19	74.286	0.000	77.469	1.094	0.523	0.001	0.497	0.012	65.222	1.747	64.285	2.361
20	74.286	0.000	77.470	1.354	0.521	0.001	0.511	0.020	65.521	1.777	61.975	1.216

Figure 4.3 Model performance without question embeddings experiments

4.6.2 Model Performance with Question Embeddings

Learning curve compare the train loss and testing loss functions. As discussed in 4.3 Loss function, learning curves help identify early stopping point. By observing above graph, we can clearly see that after 12th epoch, test loss has a increasing trend and training loss decrease further. This divergence indicates overfitting. The model is becoming too specialized in predicting the training data, resulting in poor generalization to unseen data (represented by the test set). Test model accuracy is 80% and ROC AUC is 80%. That means model can predict students answers correctness correctly 80 times out of 100 times. An ROC AUC score of 80% suggests that the classifier has a relatively good ability to distinguish between the two classes. Specifically, it means that if you randomly select one positive and one negative observation, the classifier will rank the positive observation higher than the negative one approximately 80% of the time.

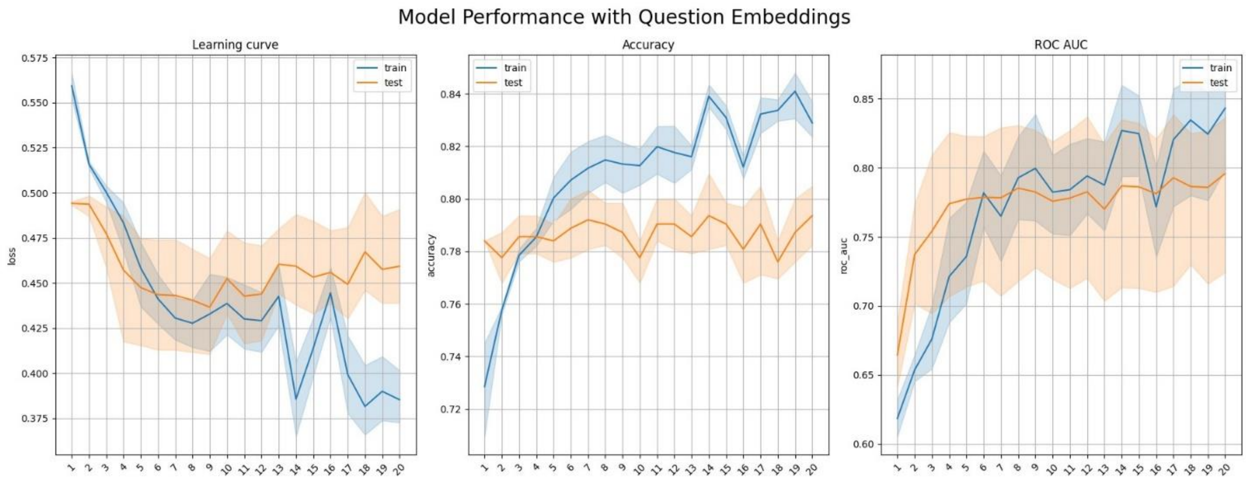

Figure 4.4 Model performance with question embeddings

	accuracy_test		accuracy_train		losses_test		losses_train		roc_auc_test		roc_auc_train	
	mean	std	mean	std	mean	std	mean	std	mean	std	mean	std
epoches												
1	78.240	0.876	77.365	1.964	0.485	0.018	0.508	0.028	74.172	5.276	67.004	2.975
2	78.880	1.073	81.780	0.679	0.450	0.007	0.432	0.012	77.407	0.772	79.320	1.338
3	78.080	0.438	82.837	0.775	0.453	0.009	0.417	0.022	79.498	2.432	81.219	2.633
4	78.080	0.912	82.920	1.223	0.440	0.007	0.390	0.028	82.263	0.556	84.280	3.188
5	79.360	0.669	77.299	2.570	0.440	0.049	0.486	0.053	80.811	6.701	72.525	8.195
6	80.160	1.043	81.321	0.708	0.402	0.016	0.409	0.006	84.467	0.901	83.038	1.441
7	80.320	1.453	83.060	0.914	0.424	0.011	0.397	0.025	83.568	0.982	84.115	1.945
8	79.200	1.697	83.110	1.225	0.437	0.011	0.382	0.027	83.154	1.257	85.023	2.846
9	77.280	0.912	76.477	4.083	0.487	0.014	0.512	0.036	73.365	5.914	66.178	4.328
10	78.080	1.073	80.094	0.451	0.445	0.005	0.457	0.010	79.902	1.004	75.428	1.294
11	79.680	0.912	81.240	1.564	0.451	0.011	0.447	0.024	78.951	1.254	77.630	2.226
12	80.000	1.131	81.799	0.845	0.450	0.010	0.413	0.034	80.832	1.372	81.970	4.167
13	78.400	0.000	76.147	3.013	0.466	0.028	0.505	0.042	76.909	4.472	68.735	6.355
14	77.920	0.438	80.256	0.888	0.424	0.014	0.427	0.012	81.919	0.997	80.875	2.257
15	78.400	0.800	82.006	0.995	0.428	0.016	0.401	0.023	83.667	0.766	83.737	2.066
16	76.640	1.043	83.125	1.276	0.447	0.006	0.382	0.025	83.000	0.471	85.098	2.498
17	78.400	0.000	77.757	2.707	0.493	0.003	0.507	0.032	65.442	1.819	66.111	2.481
18	78.560	1.640	82.495	0.426	0.495	0.004	0.445	0.009	66.337	1.629	73.471	1.465
19	78.560	0.358	83.205	1.009	0.504	0.009	0.438	0.018	64.674	1.021	75.004	2.133
20	78.880	1.339	83.833	1.155	0.516	0.012	0.433	0.019	64.944	1.246	73.012	1.762

Figure 4.5 Model performance with question embeddings experiments

5 Conclusion and Future work

This study presented a novel Deep Knowledge Tracing to predict students' ability to predict provide correct answer. Existing Deep Knowledge Tracing methods used question similarity mechanism to find the most similar questions seen by the students. This process increases computational complexity and requirement. But our study considers only most recently used questions (and learning objectives) to predict students' mastery level.

We conducted two sets of experiments. First set of experiments consider only learning object embedding to predict students' mastery level. Second experiment consider both learning object embeddings and

Comparing Accuracy and ROC AUC of both models, model with questions embeddings has a higher Accuracy and ROC AUC compared to model without question embeddings. Hence, we can conclude that question embeddings improve the model performance.

In the proceeding studies we will use this work to build question recommendation engine. Proposed methodology will be used to assess individual student's mastery level and new questions can be suggested from the question bank to increase the mastery level, adapting to student

6 Refence

Abdelrahman, G., Wang, Q., & Nunes, B. (2023). Knowledge Tracing: A Survey. *ACM Computing Surveys*, *55*(11), 1–37. https://doi.org/10.1145/3569576

Afini Normadhi, N. B., Shuib, L., Md Nasir, H. N., Bimba, A., Idris, N., & Balakrishnan, V. (2019). Identification of personal traits in adaptive learning environment: Systematic literature review. *Computers and Education*, *130*, 168–190. https://doi.org/10.1016/j.compedu.2018.11.005

Albert T. Corbertt, & John R Anderson. (1994). Knowledge Tracing : Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*.

Borges, G., & Stiubiener, I. (2014). Recommending Learning Objects Based on Utility and Learning Style. *Opening IEEE-Innovations and Internationalization in Engineering Education* .

Castañeda, L., & Selwyn, N. (2018). More than tools? Making sense of the ongoing digitizations of higher education. In *International Journal of Educational Technology in Higher Education* (Vol. 15, Issue 1). Springer Netherlands. https://doi.org/10.1186/s41239-018-0109-y

Clark, R. M., Kaw, A. K., & Braga Gomes, R. (2022). Adaptive learning: Helpful to the flipped classroom in the online environment of COVID? *Computer Applications in Engineering Education*, *30*(2), 517–531. https://doi.org/10.1002/cae.22470

Duval, E., Klamma, R., & Wolpers, M. (2007). LNCS 4753 - Personalized Links Recommendation Based on Data Mining in Adaptive Educational Hypermedia Systems. In *LNCS* (Vol. 4753).

Ennouamani, S., & Mahani, Z. (2018). An overview of adaptive e-learning systems. *2017 IEEE 8th International Conference on Intelligent Computing and Information Systems, ICICIS 2017*, *2018-January*, 342–347. https://doi.org/10.1109/INTELCIS.2017.8260060

F. M. Lord, M. R. Novick, & Allan Birnbaum. (1968). *SOME LATENT TRAIT MODELS*.

Hemmler, Y. M., & Ifenthaler, D. (2022). Indicators of the Learning Context for Supporting Personalized and Adaptive Learning Environments. *Proceedings - 2022 International Conference on Advanced Learning Technologies, ICALT 2022*, 61–65. https://doi.org/10.1109/ICALT55010.2022.00026

Hsu, P.-S. (2012). LEARNER CHARACTERISTIC BASED LEARNING EFFORT CURVE MODE: THE CORE MECHANISM ON DEVELOPING PERSONALIZED ADAPTIVE E-LEARNING PLATFORM. In *TOJET: The Turkish Online Journal of Educational Technology* (Vol. 11, Issue 4).

JanMartin Lowendahl, TerriLynn B, Thayer, & Glenda Morgan. (2016). *Top 10 Strategic Technologies ImpactingHigher Education in 2016*. http://www.gartner.com/analyst/26873

Katsaris, I., & Vidakis, N. (2021). Adaptive e-learning systems through learning styles: A review of the literature. *Advances in Mobile Learning Educational Research*, *1*(2), 124–145. https://doi.org/10.25082/AMLER.2021.02.007

Liu, M., McKelroy, E., Corliss, S. B., & Carrigan, J. (2017). Investigating the effect of an adaptive learning intervention on students' learning. *Educational Technology Research and Development*, *65*(6), 1605–1625. https://doi.org/10.1007/s11423-017-9542-1

Liu, Y., Yang, Y., Chen, X., Shen, J., Zhang, H., Yu, Y., Jiao, S., & University, T. (2020). *Improving Knowledge Tracing via Pre-training Question Embeddings*.

Martin, F., Chen, Y., Moore, R. L., & Westine, C. D. (2020). Systematic review of adaptive learning research designs, context, strategies, and technologies from 2009 to 2018. *Educational Technology Research and Development*, *68*(4), 1903–1929. https://doi.org/10.1007/s11423-020-09793-2

Mousavinasab, E., Zarifsanaiey, N., R. Niakan Kalhori, S., Rakhshan, M., Keikha, L., & Ghazi Saeedi, M. (2021). Intelligent tutoring systems: a systematic review of characteristics, applications, and evaluation methods. In *Interactive Learning Environments* (Vol. 29, Issue 1, pp. 142–163). Routledge. https://doi.org/10.1080/10494820.2018.1558257

Nakagawa, H., Iwasawa, Y., & Matsuo, Y. (2019). *GRAPH-BASED KNOWLEDGE TRACING: MODELING STUDENT PROFICIENCY USING GRAPH NEURAL NET-WORK*.

Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L., Sohl-Dickstein, J., University, S., & Academy, K. (2015). *Deep Knowledge Tracing*.

Richard Felder, by M. (2002). *LEARNING AND TEACHING STYLES IN ENGINEERING EDUCATION*. http://www.ncsu.edu/felder-public/ILSpage.html

Serra, G., & Niepert, M. (2023). *L2XGNN: Learning to Explain Graph Neural Networks*.

Song, X., Li, J., Lei, Q., Zhao, W., Chen, Y., & Mian, A. (2022). *Bi-CLKT: Bi-Graph Contrastive Learning based Knowledge Tracing*. http://arxiv.org/abs/2201.09020

Tong, S., Liu, Q., Huang, W., Huang, Z., Chen, E., Liu, C., Ma, H., & Wang, S. (2020). Structure-based knowledge tracing: An influence propagation view. *Proceedings - IEEE International Conference on Data Mining, ICDM*, *2020-November*, 541–550. https://doi.org/10.1109/ICDM50108.2020.00063

Ye, Z., Kumar, Y. J., Sing, G. O., Song, F., & Wang, J. (2022). A Comprehensive Survey of Graph Neural Networks for Knowledge Graphs. *IEEE Access*, *10*, 75729–75741. https://doi.org/10.1109/ACCESS.2022.3191784