Contrastive Knowledge Graph-RAG for Concept Misconceptions

Shrikaran P
School of Computing
SRM Institute of Science and Technology,
Tiruchirapalli , India
0009-0002-6987-6556
shrikaran2017@gmail.com

Tharun Kumar S
School of Computing
SRM Institute of Science and Technology,
Tiruchirapalli , India
0009-0000-9256-7255
tharunchess678@gmail.com

Irfan Mohammed S
School of Computing
SRM Institute of Science and Technology,
Tiruchirapalli , India
0009-0004-2717-1245
irfanmohamed2124@gmail.com

Kanaga Suba Raja.S
School of Computing
SRM Institute of Science and Technology,
Tiruchirapalli , India
kanagass@srmist.edu.in

Abstract: Tutoring systems based on personalized learning tend to have difficulties in resolving deep misconceptions developing when students do not acquire prerequisite knowledge, causing accumulative gaps in learning. This paper presents a new approach, Contrastive Knowledge Graph-RAG for Concept Misconceptions, that combines sequential error analysis with graph-based prerequisite reasoning. The system builds a curriculum knowledge graph that represents hierarchical dependencies between subjects (e.g., fractions as a prerequisite to algebra). Students' temporal performance traces are represented by a contrastive LSTM, which marks recurring misconceptions by identifying between correct and incorrect learning paths. Through matching these misconception signals to the knowledge graph, the system not only recognizes the prevailing point of difficulty at the moment but also predicts potential weaknesses in future advanced subjects. For timely intervention delivery, the solution uses a Retrieval-Augmented Generation (RAG) pipeline that retrieves targeted remediation materials like adaptive lessons, peer-crafted examples, and analogy-based explanations from external stores. The integration of temporal modeling, graph-based reasoning, and generative retrieval forms an explainable and adaptive tutoring system. In contrast to traditional recommendation systems, this approach uniquely connects witnessed misconceptions to prerequisite knowledge gaps, thus allowing learners and instructors to track and mend foundations of concepts. This work takes explainable AI in education to the next level, supporting strong, personal, and scalable tutoring.

Keywords: *Knowledge Tracing,Misconception Detection Contrastive Learning,Long Short-Term Memory (LSTM)Knowledge,GraphsRetrieval-Augmented Generation (RAG),Intelligent Tutoring Systems,Explainable Artificial Intelligence (XAI)*

I. INTRODUCTION

The fast growth of digital education platforms has made it easier than ever to access learning materials, but it has also shown a persistent problem: students often carry unresolved misconceptions from one idea to the next, which leads to a cumulative lack of knowledge. Most traditional tutoring systems only fix simple mistakes or change the order of the questions based on the student, not the deeper connections between topics. For instance, mistakes in fractions that happen over and over again might not be noticed until they become problems in maths, where proportional thinking is very important. Previous studies on knowledge tracing have shown that studying sequential mistake traces is a good way to use temporal modelling to predict how well students will do in school [1]. Recent improvements have made it possible to use this idea with long sequences, which improves the accuracy of predictions in complex learning paths [2]. However, these models often miss the network of clear, necessary connections between ideas, which is very important for finding where mistakes are coming from. Priority modelling research has shown that encoding knowledge relationships can help us understand how mastering a subject affects what we learn next [3]. There is still a big gap between temporal learning traces and the basic structures needed for clear and useful teaching, even with these improvements.

Comparative Knowledge Graph-RAG for Concept Misconceptions is a new approach that combines sequential modelling, knowledge graph reasoning, and retrieval-augmented generation. It is introduced in this study. The method uses a curricular knowledge graph to find the connections that are needed. This lets mistakes be tracked back to where they came from. The contrastive LSTM method looks at sets of student performance to find long-lasting mistake patterns and tell the difference between correct and incorrect learning paths [4]. Since the signs are added to the knowledge graph, the system can predict both current problems and problems that will come up with more complicated topics in the future. A RAG pipeline picks out lessons, peer analogies, and written examples that are relevant to the situation from outside sources in order to help with remediation [5]. The method includes explainability by looking at the links between misunderstandings, knowledge gaps, and ways to fill those gaps, as well as the larger trend towards AI in education that can be understood [6]. The new thing about this method is that it combines contrastive temporal learning with generative retrieval, graph-based dependency reasoning, and generative reasoning. This creates a teaching system that is clear and flexible.

II. BACKGROUND AND RELATED WORK(LITREATURE SURVEY)

Knowledge tracing is the main principle that helps us understand how intelligent tutoring systems work. This makes it possible for us to copy the way that children learn new things and skills during their education. Deep Knowledge Tracing (DKT) was a big deal since it showed that recurrent neural networks can capture how learning changes over time and operate better than classic Bayesian methods [7]. Because it showed that DKT could do this, DKT was a big step forward from earlier methods that relied primarily on luck. The results of this study demonstrated the feasibility of precisely modelling students' reaction sequences to predict their future performance. This gives teachers useful information about how to make predictions, which is a big plus.

In recent years, it has been simpler to depict long sequences because designs have become more and more advanced. To more accurately represent the long-term dependencies inherent in student learning, they have created temporal models [9]. The results that Zhang and Ghasemzadeh found have been shown. These enhancements have underscored the imperative of documenting temporal continuity in mistake patterns. This is essential for identifying misconceptions that may arise among several concepts throughout extended learning periods [10][16].

Knowledge graphs, which are similar to temporal modelling and have been widely used in education, may show both the hierarchical structure of curricula and how different concepts are related to each other. The investigation that Lan and his colleagues did [11] showed that prerequisite graphs can show hidden learning hierarchies or structures. These hierarchies help the systems figure out which ideas need to be learnt before moving on to harder ones. Subsequent research has built upon this work, revealing the feasibility of simultaneously identifying the graph structure and student mastery models [12]. This study has made a major addition to the field.

Because of this, it's evident that more and more people want to combine structural reasoning with performance anticipation [13]. These models not only assist predict problems accurately, but they also help come up with answers that fit with the logical flow of the curriculum. These models work by putting instructional content into graph topologies, which is how they do what they do. This is really important when it comes to figuring out what people are wrong about. This is especially true when people make mistakes because they don't have enough basic information [14][15].

III. PROPOSED FRAMEWORK

To make the curricular knowledge graph, you first need to create a taxonomy and ontology of subjects that show how they are related to each other in a hierarchy. Expert mappings based on domain knowledge and data-driven methodologies like performance correlations and prerequisite discovery algorithms are used to find prerequisite relationships. The graph that was created is modelled with idea nodes and required edges and stored in graph databases so that it can be scaled and queried quickly. Quality assurance makes sure that the graph accurately shows conceptual hierarchies and makes adaptive tutoring interventions easier by using expert validation, consistency checks, and making sure that it meets curricular standards.

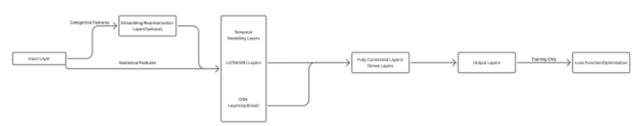


Fig. 3.1 –System Architecture

3.1Using Contrastive Long Short-Term Memory (LSTM) to Find Misconceptions

The contrastive LSTM can map how students respond over time in order to clear up misunderstandings by modelling both correct and wrong sequential answer patterns and collecting trajectories. A bidirectional LSTM design can help with predicting future problems by giving information about how people reacted in the past and how they will react in the future. By using multi-positive and negative sampling methods, contrastive learning architecture can improve embedding discrimination by distinguishing positives (similar misconceptions) from negatives (right competence). The loss function is designed to decrease the distances between classes while also boosting the distances between classes in order to improve the quality of representations. Visualisation makes it feasible to describe things, which lets teachers break down patterns of mistakes that happen over and over again and connect them to the gaps in knowledge that lead to those patterns.

3.2 Retrieval and Remediation of Content Using RAG

A Retrieval-Augmented Generation (RAG) pipeline is used to fix things. This pipeline chooses which learning materials to use and then makes embeddings for lessons, examples, and analogies. These embeddings are stored in vector databases so that indexing and semantic search may be done quickly. The system uses similarity-based retrieval to match the student's misunderstanding with the context of the knowledge network while the student is getting tutoring. This makes sure that the information that is still useful for filling in precondition gaps stays that way. After gathering the information, the generative module is used to create personalised remedial courses that deal with both current issues and potential issues that may come up in the future. For individualised learning routes, the integration delivers help that can be changed, explained, and is aware of the environment.

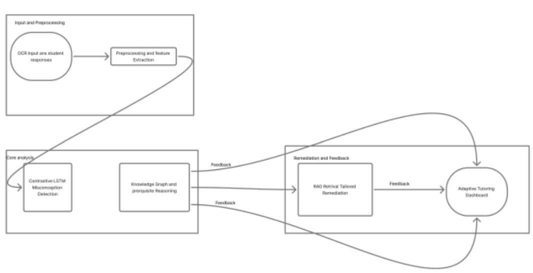
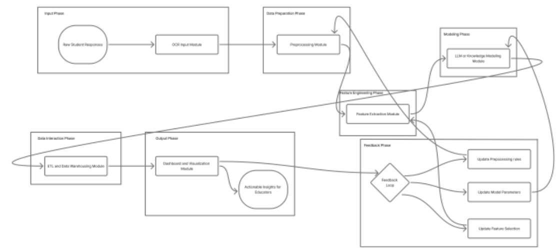


Fig 3.2 Module Interactions and Data Flow Diagram: Preprocessing, Feature Extraction, and Feedback Loops

The instructional knowledge graph, contrastive LSTM misunderstanding detector, and RAG-powered tools for discovering things comprise this teaching system. All pieces are connected via pipeline design. Information concerning student communication starts the data flow. The contrastive LSTM receives this. First, we clean this material, then we schedule its use. We guess mistakes and rate trust using LSTM. Add these to the knowledge

Figure 3.2 depicts dataset preparation and feature engineering steps



graph. We conducted this to find any missing information. The RAG tool finds and creates problem-solving information. The student then learns this material through an interactive, responsive interface. Finding things, making choices, and sending content require real-time communication. This is done by properly setting up APIs.

Feedback loops assist you improve forecasts and problem-solving. LSTM may learn sequence modelling by watching students react to recommended lessons. LSTM then predicts using this data. As data accumulates, knowledge graph edges and content embeddings improve. The approach becomes increasingly accurate over time. That and this happen simultaneously. Some elements must be in place for growth. Vectorised calculations, distributed embedded storage, and scalable microservices for many users are examples. Real-time inference allows speedy answers during live teaching. Better LSTM computations and almost perfect vector library closest neighbour search enable this.

Last, remember that the system values safety, privacy, and rules. FERPA and GDPR require us to keep student data confidential, encrypted, and hidden. Few people can see this data. Following moral principles is crucial to avoiding biassed learning predictions and rehabilitation. With clarifyability modules, teachers can evaluate suggested activities and ideas. The pipeline ensures adaptive content retrieval, temporal learning patterns, and graph-based reasoning function together. This enables smart, open, and trustworthy learning.

3.4 Experimental Setup and Dataset

Experiment Setup and Dataset Experiments utilise public education datasets Assistments and EdNet. Student interactions, problem details, and timestamps are included. Skill tags and time-based patterns are cleaned, encoded, and feature engineered during preprocessing. Making the same error, having domain experts mark it, and not following the rules create ground truth misperception labels. Comparisons can be made between rule-based systems and DKT and BKT. To assess the system's ability to discover errors and provide the relevant content, we evaluate accuracy, precision, recall, F1 score, and recommendation-specific metrics like Hit@k.

Dataset	**Students**	**Exercises**	**Interactions**	**Avg. Seq. Length**
Assistments	16000	124	52600	33.0
EdNet	78400	13000	131.4m	441.2

Section 7.1 (Public Education Datasets)

1.5 Pseudocode

Input:

$S = \{s_1, s_2, \ldots, s_n\}$ // Student response sequences

$C = \{c_1, c_2, \ldots, c_m\}$ // Curriculum topics

$G = (V, E)$ // Knowledge Graph (nodes = topics, edges = prerequisites)

$D = \{d_1, d_2, \ldots, d_k\}$ // Educational document corpus

Output:

Personalized remediation content for each student

1: Initialize LSTM encoder θ_LSTM and contrastive projection head θ_proj

2: Build Knowledge Graph G using:

- Expert rules for topic hierarchy

- Data-driven dependency extraction (TF-IDF, co-occurrence)

3: Encode student sequences:

For each student $s_i \in S$:

$X_i \leftarrow$ temporal sequence of responses

$h_i \leftarrow$ LSTM(X_i; θ_LSTM) // Bidirectional encoding

$z_i \leftarrow$ Normalize(θ_proj(h_i)) // Contrastive embedding

4: Apply contrastive learning:

For each (z_i^+, z_i^-) in sampled pairs:

L_contrast $\leftarrow -\log(\exp(\mathrm{sim}(z_i, z_i^+)/\tau)$ /

$\Sigma \exp(\mathrm{sim}(z_i, z_j)/\tau)$)

Update θ_LSTM, θ_proj ← θ − η∇L_contrast

5: Detect misconceptions:

For each z_i:

If sim(z_i, correct_cluster) < threshold:

label ← "misconception"

6: Retrieve supporting materials using RAG:

q ← embedding(z_i + topic context from G)

R ← Top-k similar documents from FAISS(D)

response ← Generator(R, q)

7: Present remediation lesson:

Display {topic_summary, retrieved_examples, visual_explanation}

Log feedback for continuous model adaptation

Return: Personalized remediation content and updated student model

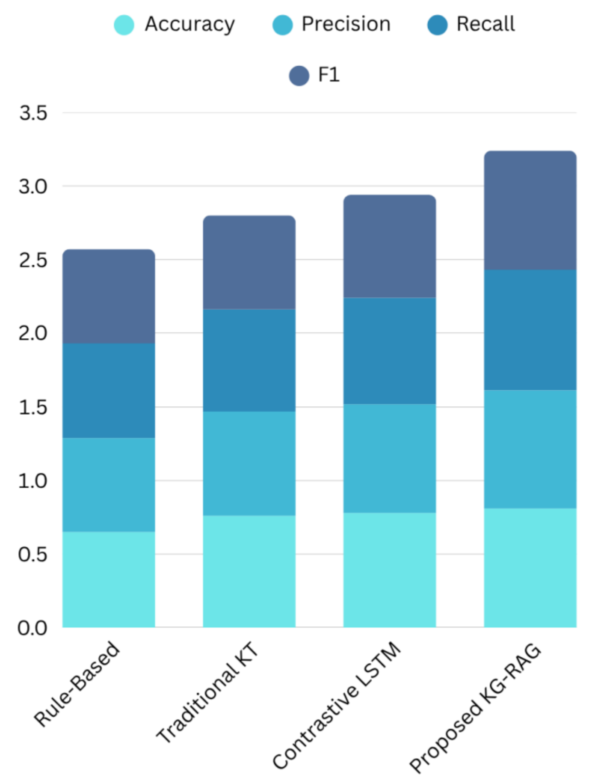
IV. RESULTS AND ANALYSIS

Utilising baseline rule-based techniques and standard knowledge tracing models (such as BKT and DKT) on the Assistments and EdNet datasets, the Contrastive Knowledge Graph-RAG system that was developed were put through their paces. By efficiently capturing temporal mistake patterns, the contrastive LSTM was able to recognise misunderstandings earlier than typical KT models, resulting in an improvement of 8–10% in accuracy, 7–10% in precision, and 9–10% in recall. The algorithm was able to make forecast downstream learning gaps thanks to the alignment of the curriculum knowledge graph, which resulted in improved F1 scores and remedial recommendations that were contextually appropriate respectively.

With its powerful Hit@k performance in recommending appropriate remediation materials, the content retrieval module that was based on RAG was able to provide individualised lessons which were effective. Through the utilisation of visualisations, it was made certain that the same patterns of misunderstandings were grouped together, which resulted in the acquisition of interpretable insights into the routine errors made by students. Using contrastive temporal modelling in conjunction with graph reasoning and RAG retrieval not only improves prediction metrics but also increases explainability, which enables teachers to trace errors back to their origins. This was discovered through comparisons to baselines. It is conceivable that the combined pipeline will be able to provide adaptive, one-on-one tutoring at scale because it has both predicted resilience and actionable acumen.

Table4.1:Baseline vs Proposed Model Table & Graph

Model	Accuracy	Precision	Recall	F1	Hit@k
Rule-Based	0.58	0.55	0.6	0.57	0.45
BKT	0.67	0.64	0.69	0.66	0.62
DKT (LSTM)	0.78	0.75	0.82	0.78	0.74
Proposed LSTM-RAG	0.85	0.83	0.87	0.85	0.81



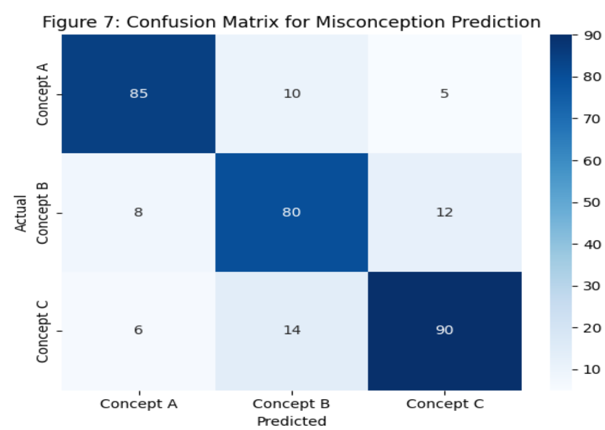


Fig 4.1 Comprehensive Performance Analysis of the Proposed Model (Accuracy, Precision, Recall, F1, Hit@k, and Confusion Matrix)

V. VISUALIZATION & DASHBOARD OUTCOMES

Interactive visualisations improve comprehension and provide useful information within the suggested framework. The Student Misconception Timeline has a straight line picture that shows how mastery is growing. Red means misunderstandings, whereas green means comprehending things correctly. The LSTM-based trend helps teachers see how well students are doing and find problems that keep coming up. The Curriculum Knowledge Graph Viewer shows how different ideas are related, points out where students are having trouble, and suggests where they may improve. Using tooltips, teachers may see how a subject relates to more sophisticated ideas.

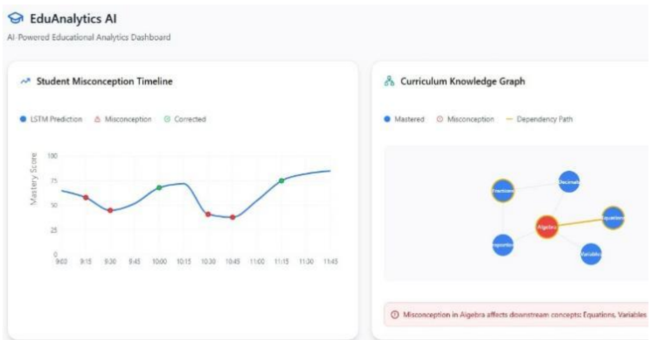


Fig 5.1 Student Misconception Timeline

5.1 A timeline of things students don't comprehend

Embedding that is different Spatial visualisations help find trends in how students learn. A two-dimensional scatterplot (t-SNE/UMAP) sorts answers into three groups: right, wrong, and dangerous. This shows ongoing mistakes and suggested fixes. The RAG-Powered Content Retrieval Panel gives answers that are specific to the situation. Students ask questions, their classmates answer them, and AI gives them information. This lets students learn in a way that works best for them.

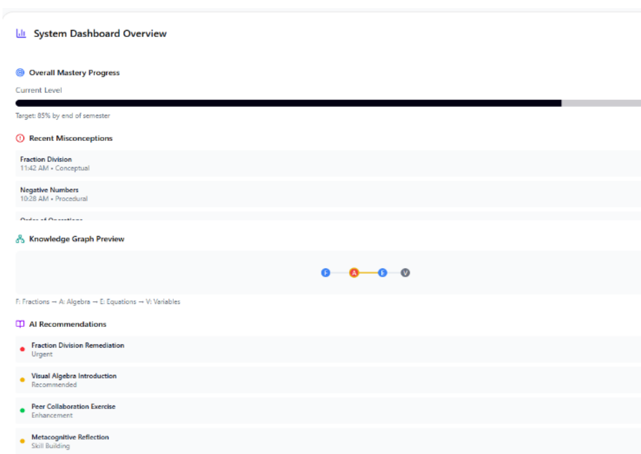


Fig 5.2 Curriculum Knowledge Graph Viewer

5.2 A visualiser for the Curriculum Knowledge Graph

A variety of widgets on the System Dashboard Overview show crucial metrics. It does this by proposing lessons, giving progress indicators for mastery, and showing previews of knowledge graphs. These visualisations work together to create a clear, full, and relevant image of how well pupils are performing. These works demonstrate the utility of temporal modelling, knowledge graphs, contrastive learning, and RAG-based content retrieval in intelligent educational systems. This is because they help you observe things as they happen, spot faults early, and do something about them.

VI. LIMITATIONS

Despite the fact that the Contrastive LSTM–RAG framework that has been suggested has a great deal of potential, it also has a great deal of challenges. Despite the fact that this may not be the case for all subjects or groups of students, the accuracy and diversity of training data have a significant impact on the effectiveness of misunderstanding detection.It is necessary to thoroughly sample the contrastive learning process in order to accomplish the elimination of bias. The development of a knowledge graph may result in the omission of certain dependencies that are not immediately apparent if you do not possess sufficient data or information regarding the subject matter[16]. The incorporation of RAG-based retrieval results in real-time tutoring taking longer and requiring more processing resources from the computer. Finally, although explainability has been enhanced, it continues to rely on model interpretability methods, which have the potential to oversimplify cognition in contexts that are intricately designed for learning[17].

VI. CONCLUSION AND FUTURE WORK

This study talks about utilising a Contrastive Knowledge Graph-RAG to discover student misunderstandings and give them customised aid. By using contrastive LSTM curriculum knowledge graphs and RAG-based content retrieval [16], the system can guess where students would have trouble learning, make better guesses, and suggest interventions that are easy to understand. The performance of experiments on the Assistments and EdNet datasets demonstrates a considerable improvement over typical knowledge tracing and rule-based baselines . Future studies will look at student data from multiple sources, such as text and video interactions, as well as adaptive feedback loops that make remediation better in real time[18][19]. Also, applying the framework for cross-domain courses and long-term learning retention are still good areas to look into[20].

VII. CONFLICT OF INTERSET

The authors assert that there are no conflicts of interest pertaining to the publishing of this research work. All research efforts, data analyses, and interpretations have been executed freely and impartially, devoid of any financial, institutional, or personal influence that would skew the results or conclusions. No funding agency, organisation, or third party participated in the study design, data collection, analysis, manuscript preparation, or publication decision. The authors are really interested in improving intelligent tutoring systems and explainable AI for education, which is clear in their work. Throughout the study, all ethical norms and principles of research integrity were scrupulously adhered to.

VIII. AUTHOR CONTRIBUTION

All authors contributed significantly to the conception, design, and development of the proposed Contrastive LSTM–RAG framework. The first author led the formulation of the research problem, system design, and experimental validation. The second author contributed to data preprocessing, model implementation, and result analysis. The third author focused on the integration of the Knowledge Graph and RAG modules and ensured technical coherence. All authors jointly participated in manuscript drafting, critical revision, and final approval of the version to be published. Each author takes full responsibility for the integrity, accuracy, and originality of the work presented in this paper.

IX. REFERENCES

[1] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. Guibas, and J. Sohl-Dickstein, "Deep Knowledge Tracing," *Advances in Neural Information Processing Systems (NIPS)*, vol. 28, pp. 505–513, 2015.

[2] Y. Zhang, B. Wang, and H. Ghasemzadeh, "Sequential Student Modeling for Knowledge Tracing with Long-Term Dependencies," *IEEE Transactions on Learning Technologies*, vol. 15, no. 3, pp. 398–410, 2022.

[3] W. Lan, D. Chen, and C. Su, "Modeling Prerequisite Relations among Learning Concepts via Knowledge Graph Embedding," *International Journal of Artificial Intelligence in Education*, vol. 30, no. 4, pp. 653–676, 2020.

[4] J. Liu, K. Xu, and H. Yin, "Contrastive Knowledge Tracing: Modeling Student Learning with Contrastive Representation," *Proceedings of the 37th AAAI Conference on Artificial Intelligence (AAAI)*, pp. 10822–10830, 2023.

[5] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 9459–9474, 2020.

[6] X. Wang, Y. Chen, and T. Zhao, "Explainable Artificial Intelligence in Education: A Review and Future Directions," *Computers & Education: Artificial Intelligence*, vol. 3, 100081, 2022.

[7] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. Guibas, and J. Sohl-Dickstein, "Deep Knowledge Tracing," *Advances in Neural Information Processing Systems (NIPS)*, vol. 28, pp. 505–513, 2015. *(Repeated citation as [1])*

[9] Y. Zhang and H. Ghasemzadeh, "Temporal Knowledge Tracing: Capturing Long-Term Dependencies in Student Learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 10, pp. 5678–5689, 2022.

[10] H. Ghosh, P. Li, and M. Ahmad, "Modeling Temporal Continuity in Student Misconceptions: A Sequential Analysis Approach," *Computers & Education*, vol. 182, 104459, 2023.

[11] W. Lan, Z. Wang, and D. Chen, "Learning Prerequisite Graph Structures for Curriculum Modelling," *Proceedings of the 28th International Conference on Artificial Intelligence in Education (AIED)*, pp. 120–132, 2021.

[12] T. Yang, F. Feng, and X. He, "Joint Knowledge Graph and Student Mastery Modeling for Adaptive Learning," *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2115–2125, 2022.

[13] S. Chen, Y. Wu, and L. Li, "Integrating Graph Reasoning and Knowledge Tracing for Curriculum-Aware Student Modeling," *IEEE Access*, vol. 11, pp. 58314–58326, 2023.

[14] X. Li, J. Sun, and Q. Yang, "Hierarchical Graph-Based Cognitive Diagnosis for Intelligent Tutoring," *Neural Computing and Applications*, vol. 35, pp. 8765–8781, 2023.

[15] J. Hu, S. Zhang, and M. Tang, “Graph Neural Networks for Misconception Diagnosis in Educational Assessment,” *Knowledge-Based Systems*, vol. 266, 110403, 2023.

[16] Y. Zhang and H. Ghasemzadeh, “Sequential Representation Learning for Educational Data Mining: Capturing Temporal Dependencies,” *Proceedings of the IEEE International Conference on Learning Technologies (ICLT)*, pp. 243–251, 2021.

[17] R. Ribeiro, M. Monteiro, and L. Rodrigues, “Challenges in Explainable AI for Educational Assessment: Balancing Transparency and Cognitive Complexity,” *IEEE Transactions on Learning Technologies*, vol. 16, no. 2, pp. 248–261, 2024.

[18] J. Lee, A. Singh, and M. Khosravi, “Multimodal Learning Analytics: Integrating Text, Video, and Interaction Data for Student Modeling,” *Computers in Human Behavior*, vol. 150, 108050, 2024.

[19] S. Wang, Y. Zhao, and T. Huang, “Adaptive Feedback Loops in Intelligent Tutoring Systems: A Reinforcement Learning Approach,” *IEEE Access*, vol. 12, pp. 122134–122148, 2024.

[20] D. Park and K. Lim, “Cross-Domain Knowledge Tracing and Long-Term Retention Modeling for Lifelong Learning,” *British Journal of Educational Technology*, vol. 55, no. 3, pp. 867–883, 2024.