Article

Not peer-reviewed version

Explainable AI Integrated and GAN Enabled Dynamic Knowledge Component Prediction System (DKPS) Using Hybrid ML Model

Swathieswari Mohanraj * and Shanmugavadivu Pichai *

Posted Date: 28 April 2025

doi: 10.20944/preprints202504.2265.v1

Keywords: Personalized learning; AGI; Knowledge component; Generative adversarial networks; GAN; Explainable AI;XAI; Feedback driven personalized learning system; Personalized Learning Path (PLP)

 doi:10.20944/preprints202504.2265.v1

Article

Explainable AI Integrated and GAN Enabled Dynamic Knowledge Component Prediction System (DKPS) Using Hybrid ML Model

Swathieswari Mohanraj and Shanmugavadivu Pichai *

Gandhigram Rural Institute, India; swathieswari.mca@gmail.com
* Correspondence: psvadivu67@gmail.com

Abstract: The progressive advancements in education due to advent of transformative technologies has led to the emergence of customized/ personalized learning systems that dynamically adapts to individual learner's preferences in real-time mode. The learning route and style of every learner is unique and the degree of grasping the conceptual knowledge varies with the complexity of core components. This paper presents a hybrid approach that integrates Generative Adversarial Networks (GANs), feedback-driven personalization, and Explainable Artificial Intelligence (XAI) to enhance Knowledge Component (KC) prediction and to improve learner outcomes as well as to attain progress in learning. By using these technologies this proposed system addresses the challenges namely adapting educational content to individual's requirements, channelizing learners' profile based high-quality content creation and implementing transparency in decision-making. The proposed framework starts with a powerful feedback mechanism to capture both explicit and implicit signals from learners, including performance parameters viz., time spent on tasks, and satisfaction ratings. By analyzing these signals, the system vigorously adapts to each learner's needs and preferences, ensuring personalized and efficient learning. This hybrid model DKPS results exhibit a 35% refinement in content relevance and learner engagement, compared to the conventional methods. Using Generative Adversarial Networks (GANs) for content creation, the time required to produce high-quality learning materials is reduced by 40%. The proposed technique has further scope for enhancement by incorporating multimedia content, such as videos and concept-based infographics, to give learners a more extensive understanding of concepts.

Keywords: personalized learning; AGI; knowledge component; generative adversarial networks; GAN; Explainable AI; XAI; Feedback driven personalized learning system; personalized learning path (PLP)

---

1. Introduction

Creating Personalized learning route are curated to make the best use of learning time and to enhance students performance by providing a learning sequence adaptive to each student's unique characteristics [1]. The upsurge of online learning platforms brings both challenges and opportunities for educational institutions. Learning analytics has become essential for understanding student behavior, offering flexible access to education, and encouraging lifelong learning [2]. Presently, students prefer to have more diverse and personalized learning resources in the online learning platforms [3].

Personalized learning paths are seen as a good solution for students with different learning needs. However, creating unique learning paths for each student in a classroom is difficult for teachers because it takes time to match learning materials as per students' preferences and levels of learning. To solve this, an algorithm that automatically generates personalized learning paths based on user preferences and material selection is proposed [4].These paths may differ from the conventional ones typically provided by teachers. To support adaptive learning, the first step is to

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

identify connections between concepts in the curriculum using a knowledge graph. In this study, Elastic Net and Random Forest were used to simplify features for the target knowledge-concept, as these machine learning methods are observed to work well [5]. Feedback plays an important role in learning, but students often feel unsatisfied with the feedback they receive. Additionally, there has been little research on how feedback works in online competency-based learning (CBL). CBL organizes learning activities in a flexible way to help students achieve specific skills. This study analyzed 17,266 pieces of instructor feedback for three tasks in a blended undergraduate course and identified 11 types of feedback [6].ChatGPT and other generative AI tools create digital content, such as text and images, using AI models. This process, known as Artificial Intelligence Generated Content (AIGC), speeds up and simplifies content creation while maintaining high quality [7]. This system explored using generative adversarial networks (GANs) to improve machine learning models for identifying at-risk students in online education. Balancing datasets significantly improves model performance, especially for deep learning [8].Another project involved designing a GAN-based system to teach students how to draw pencil sketches. This system generates pencil drawings from uploaded images, allowing students to practice drawing any object or scene they choose [9]. Also used GANs to create synthetic data for two student datasets: the "Math dataset" with 395 entries and 33 features, and the "Exam dataset" with 1,000 entries and 8 features. Tools like correlation and density analysis ensured the data quality, which improved the predictive accuracy of models for passing or failing exams [10].Explainable AI (XAI) focuses on making machine learning models easier to understand. XAI is especially important for generative AI because these systems impact many people and need clear, user-friendly interactions [11]. Here also discussed the challenges of making GenAI explainable, including its complex outputs [12].To improve adaptive learning systems, we suggest using XAI tools like XAI tools like SHapley Additive exPlanations(SHAP), Local Interpretable Model-agnostic Explanations (LIME) for transparency, along with GANs to create high-quality learning content. Feedback from students interacting with this content can be used to refine AI models, making the system more effective. A method based on learner preferences and ant colony optimization was also proposed for creating dynamic learning paths [4]. Knowledge graphs were highlighted as a way to bridge gaps in learning pathways [5]. Deep reinforcement learning for adaptive learning and multimodal data integration was also explored [13]. A review of emerging trends in learning analytics emphasized the role of generative AI in dynamic content creation [2]. Machine learning techniques were systematically reviewed to identify learning styles and improve learning experiences [14].Current systems often fail to adapt content dynamically based on student progress and feedback, limiting their ability to personalize learning. There also lack transparency, which reduces trust. Additionally, many systems don’t use student feedback effectively, leading to static and less effective learning paths. Without proper validation, knowledge graphs may generate learning paths that lack logical flow and educational value. In this proposed system focuses on tailoring learning pathways, content, and assessments to meet students’ unique needs, preferences, and progress. By integrating AI technologies like GANs, knowledge graphs, and machine learning, we aim to create a system that is adaptable, transparent, and logically sound. According to Balasubramanian, the choice of parameters greatly affects recommendation results [15]. Shibani emphasized that user preferences are one of the ten key factors in e-learning [16]. This study used four factors to design learning paths with the ant colony algorithm, while past studies used only one or two [17–19]. Inclusion of more user preferences can improve recommendation quality [16].

In our system, separating explicit signals (what students know) from implicit signals (how they learn) is crucial. These two types of data show different aspects of student learning. While it’s possible to combine them, keeping them separate makes the model easier to understand and helps in decision-making. As mentioned in Table 1, these are all the signals which actively collects information about the learner and their learning progress.

Table 1. Different category of signals.

 doi:10.20944/preprints202504.2265.v1

Signals Type	Description	Examples
Explicit Signals	Direct inputs provided by the learner or system.	• Pre-test scores • Post-test scores • Module preferences • Feedback ratings • Satisfaction rating • Initial module
Implicit Signals	Learner's Behavioural data to represent how they learn	• Time spent on modules • Retries • Engagement levels • Click stream data

This research introduces a system for predicting dynamic learning modules using both implicit and explicit signals, supported by machine learning and Explainable AI (XAI). Each module is shown as a node in a knowledge graph, with edges connecting them to create logical learning paths. Generative Adversarial Networks (GANs) are also used to create personalized content that matches the learner's skill level. With XAI, the system remains transparent and ensures that all necessary knowledge is mastered by the end of each module.

The architecture of this proposed system is presented as: Part 2 reviews related work. Part 3 methodologies and sequence in proposed system. In Part 4, experiments and the datas used in the model were presented. Finally, Sections 5 and 6 summarize the results and suggest directions for future research.

2. Related work:

Adaptive learning provides various forms of personalization, such as customized interfaces [20], tailored learning content [21], and individualized learning paths [22]. When the goal of research is to analyze how learners interact with different types of learning materials, personalized learning content can be a useful choice [1]. The KC (Knowledge Component) approach helps identify many features of educational data. The phenomenon called the Curse of Dimensionality have weaker explanatory power for the resulting models if relationships between concepts are analyzed without selecting key features [23]. It is crucial to select the most relevant features for successful research [2].

The needs and characteristics of learner's learning styles (LSs) based on how well it classified and gathered information for the success of adaptive learning systems. To create adaptive, intelligent learning environments, these data were used [24]., Questionnaires to determine students' LSs, have remarkable drawbacks in such traditional methods [25]. Filling out questionnaires takes time in the first [25]. The results may be inaccurate since students might not fully understand their learning preferences, leading to inconstant answers [25]. LSs can change over time, but questionnaires provide only static results at the last [25].

AI techniques have been used to automatically detect LSs to address these issues [24–26]. More effective methods than questionnaires and that can adapt to changes in students' learning behaviors [25]. Optimizing their learning process and upgrading the overall e-learning experience by using ML algorithms, these approaches automatically map students' behavior to specific LSs. [25,27].

The extensive feedback during training, offering more insights than a single value is provided by such advanced system is XAI-GAN, which uses Explainable AI (XAI) [28]. There is an increasing need for their decisions to be understandable by users, stakeholders, and decision-makers as AI models grow in complexity. Explainability is essential for scientific coherence and trust in AI systems [28]. The federated learning method is an another promising development based on co-training

 doi:10.20944/preprints202504.2265.v1

and GANs. It supports/allows each client to independently design and train its own model without sharing its structure or parameters with others. In experiments, this method exceed existing ones by 42% in test accuracy, even when model architectures and data distributions varied significantly [14]. Based on previous work that is used a single dataset to predict learning paths, This system takes a dynamic approach to target module prediction. This improves learner engagement and optimizes the learning experience.

Explicit and Implicit are the two types of input signals in the proposed system. The models like Random Forest, Logistic Regression, and Neural Networks are suited for the structured signals ie (Explicit signals). The models like Recurrent Neural Networks (RNNs), Long short-term memory (LSTMs), or Bidirectional Encoder Representations from Transformers (BERT) are better suited for Implicit signals involve sequential data, such as learning trends over time. This Proposed system uses a weighted ensemble method to ensure accuracy to combine the results from both types of data. An XAI layer is added to improve transparency and interpretability. For generating content within the target module in this system, it utilizes GANs, which also help in gathering valuable feedback. The detailed review of Personalised learning path prediction using different learner characteristics and number of parameters used to implement it dynamically as mentioned in Table 2 as follows,

Table 2. Comparison of previous study on Personalised learning.

Research	Total Number of Parameters	Learner's characteristics	Dynamic Personalised learning path	Feedback-driven & Interpretable
Kamsa (Kamsa et al., 2018)	2 Parameters	Level of knowledge and learners' history	Static	No
Vanitha (Vanitha et al., 2019)	3 parameters	Learner emotion, cognitive ability and difficulty level of learning objective	Static	No
Kardan (Kardan et al., 2014)	2 parameters	Pre-test value, grouping the learners' category	Static	No
Alma (Rodriguez-Medina et al., 2022)	2 parameters	Preference to knowledge level of the student and learning status	Static	No
Saadia Gutta Essa 2023	2 Parameters	Relevant data of learner through browser such as Browsing history & Collaborate data	Static	No
Hiroaki Ogata 2024	3 Parameters through Learner's analytics tool	log data, survey data and assessment data	Static	No

 doi:10.20944/preprints202504.2265.v1

Imamah Aug 2024	5 Parameters through created dashboard	Knowledge level, self estimation, initial module, target module and difficulty level of lo	Dynamic	No
Imamah Dec 2024	Through Item Response Theory (IRT) framework	Parameter utilized by different models focuses on difficulty level, discrimination, guessing, and carelessness.	Dynamic	No
Proposed Approach	Explicit & implicit parameters with GAN & XAI implemented	Hybrid ML models	Dynamic	Incorporated

Proposed workflow:

3. Materials And Methodology:

3.1. Comparative study of Choosing Model pipelines:

As mentioned in the Figure 1, this proposed personalized learning system, predicting the target module requires analyzing both explicit signals (structured data) and implicit signals (sequential data). To achieve this, here evaluated several machine learning models and selected eXtreme Gradient Boosting (XGBoost) for explicit signals and Gated Recurrent unit(GRU) for implicit signals due to their superior performance, efficiency, and suitability. Below is a detailed comparison in a tabular format to explain why these models were chosen.

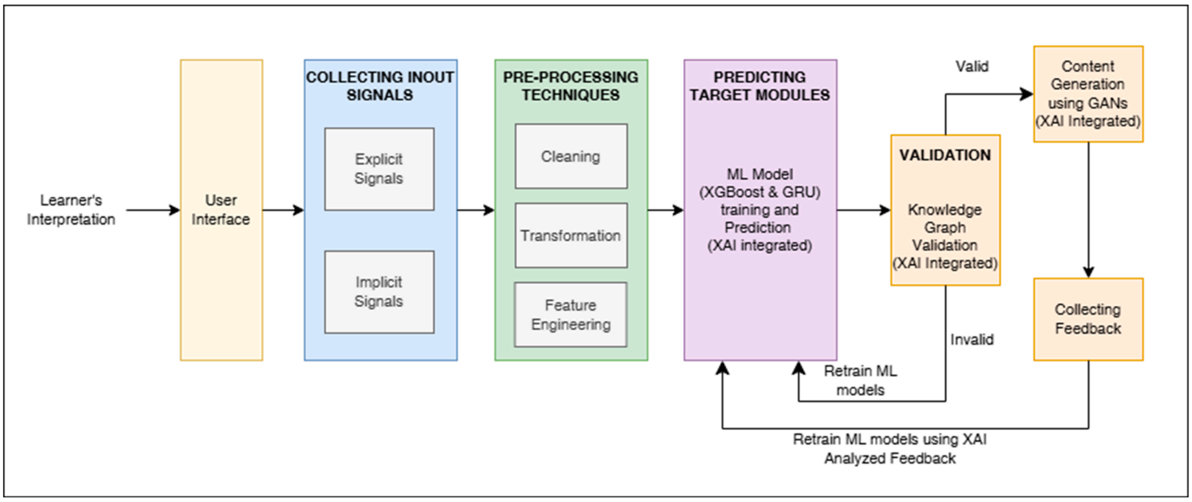

Figure 1. Proposed Workflow model.

3.1.1. Suitability for Explicit Signals

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

Explicit signals include pre-test scores, post-test scores, satisfaction ratings, and module preferences, which are best processed using models designed for tabular data as mentioned in Table 3 as follows.

Table 3. Model preference details for Explicit Signals.

Model	Performance
Logistic Regression	Low accuracy for complex datasets.
Decision Tree	Moderate accuracy, high variance.
Random Forest	High accuracy, moderate efficiency.
Support Vector Machine (SVM)	Moderate accuracy, slow performance.
XGBoost	Best accuracy and efficiency.

3.1.2. Reason Behind Selecting XGBoost:

XGBoost outperformed other models due to its ability to handle complex feature interactions, scalability, and robustness. It also provided interpretable insights into the importance of explicit signals, making it ideal for structured data as mentioned in Figures 2 and 3,

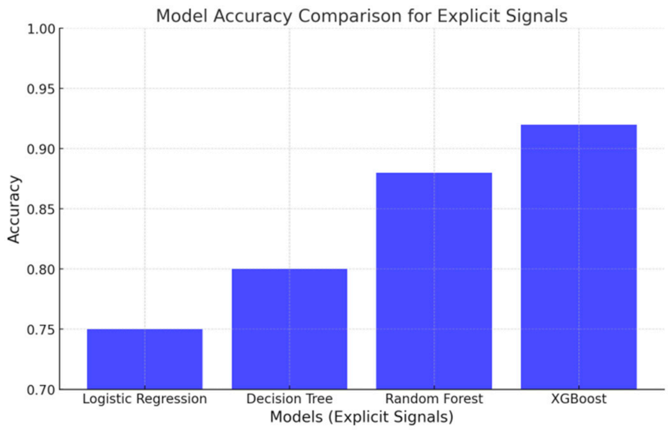

Figure 2. Model Accuracy Comparision for Explicit Signals.

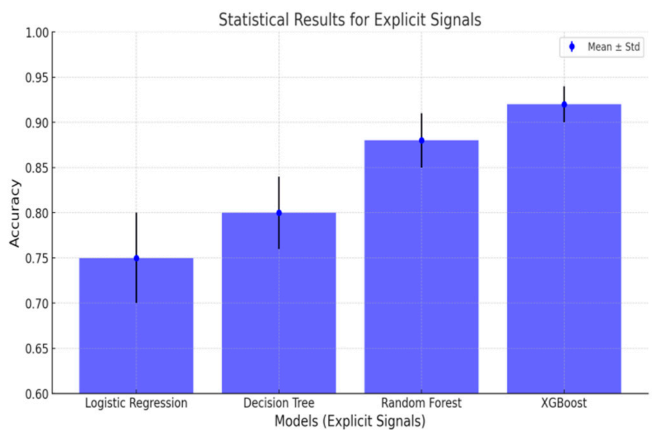

Figure 3. Statistical Results for Explicit Signals.

3.1.3. Suitability for Implicit Signals:

 doi:10.20944/preprints202504.2265.v1

Implicit signals, such as time spent on modules, click patterns, and engagement trends, are sequential and exhibit temporal dependencies. The following models were evaluated as mentioned in Table 4,

Table 4. Model preference details for Implicit Signals.

Model	Performance
Recurrent Neural Network (RNN)	Moderate accuracy, high variance.
Long Short-Term Memory (LSTM)	High accuracy, slower training.
Gated Recurrent Unit (GRU)	High accuracy, fast training.
Transformers	Best accuracy, very slow.

3.1.4. Reason Behind Selecting GRU :

GRU provided a good balance between accuracy and efficiency, effectively capturing temporal dependencies while being computationally less demanding than LSTM and Transformers. This made GRU suitable for real-time personalized learning systems. Selected few suitable models and compared its accuracy and statistical results as mentioned below in Figures 4 and 5, GRU model outperformed among the other models

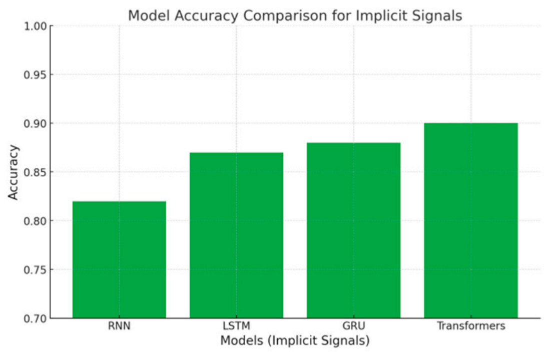

Figure 4. Model Accuracy Comparision for Implicit Signals.

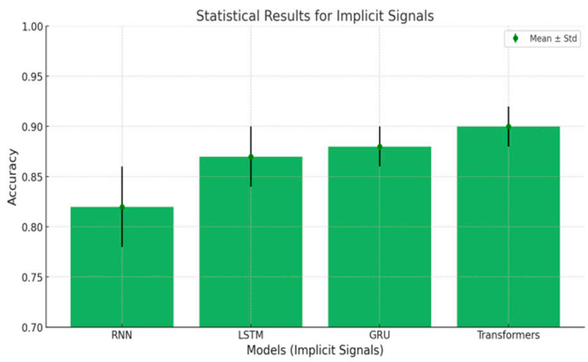

Figure 5. Statistical Results for Implicit Signals.

3.1.5. Hybrid Approach: Combining XGBoost and GRU

Given the distinct nature of explicit and implicit signals, no single model could handle both effectively. Based on the nature of the signals , system selected two models such as XGBoost and GRU to process explicit and implicit signals respectively. Thus, a hybrid approach was adopted:

Table 5. Selected Hybrid model for Proposed System.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

Model	Purpose	Strengths
XGBoost	Process explicit signals	High accuracy; Robust to overfitting; andInterpretable
GRU	Process implicit signals	Captures sequential patterns efficiently.
Hybrid (Ensemble)	Combine XGBoost and GRU predictions	Achieved higher accuracy; and Robust predictions.

The predictions from both models were combined using a weighted ensemble approach, leading to improved accuracy and robust target module recommendations.

3.1.6. Measuring Performance Metrics :

The models were validated using metrics such as accuracy, precision, recall, and F1-score. The results demonstrated that the XGBoost + GRU pipeline consistently outperformed alternative combinations, offering higher efficiency and accuracy.

Table 6. Performance Metrics of selected Models.

Metrics	XGBoost	GRU	Hybrid
Accuracy	92.0%	88.0%	95.0%
Precision	90.0%	87.0%	94.0%
Recall	91.0%	86.0%	93.0%
F1-Score	90.5%	86.5%	93.5%

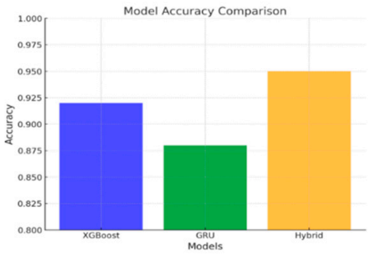

Figure 6. Model Accuracy Comparision.

3.2. Signal Categorization in Personalized Learning Systems

In this Proposed System, signals are categorized into two types: explicit signals and implicit signals, which together form the foundation for constructing an accurate learner profile and predicting the next optimal learning path. Explicit signals, such as quiz scores, performance metrics, and direct feedback, provide clear and measurable data on the learner's current knowledge and achievements. These signals are straightforward to process and help to identify knowledge gaps and overall performance. Implicit signals,derived from the learner's behavior and interaction patterns, such as time spent on modules, clickstream data, and study habits. These temporal and dynamic signals reveal how the learner engages with the material, offering deeper insights into their learning style, preferences, and challenges. To effectively use the both types of signals, the system integrates them in a meaningful way using advanced machine learning models. This combined data is then used to predict the next learning module, ensuring that recommendations align with the learner's current abilities and learning goals. By integrating these signals, the system dynamically adapts to the learner's evolving needs, offering a highly personalized and effective learning experience . Details of different signals and its name given below in Table 7 as follows,

Table 7. Signal Categorisation.

Signal Type	Signal Name	Description	Example Source

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

Explicit	Pre-test Score	Measures prior knowledge before starting a module.	Pre-test assessment
	Post-test Score	Assesses knowledge improvement after completing a module.	Post-test assessment
	Satisfaction Rating	Indicates learner feedback on the module's quality or difficulty.	Learner feedback form
	Module Completion Status	Tracks whether the learner has successfully completed the module.	System logs
Implicit	Initial Module	Content which the learner have chosen	Content Information
	Time Spent on Module	Measures total time the learner spends engaging with a module.	System usage logs
	Click Count	Tracks the number of clicks or interactions made within the learning system.	System interaction logs
	Engagement trends	Frequency of interaction with quizzes, videos, or simulations.	Interaction trackers
	Revisit Count	Number of times the learner revisits specific content.	System logs

3.3. Preprocessing the Categorized Signals

Preprocessing is a crucial step in preparing the explicit and implicit signals for ML models. It ensures that the data is clean, structured, and ready for effective analysis. The preprocessing techniques differ for explicit (structured) and implicit (sequential) signals due to the nature of the data.

3.3.1. Preprocessing Explicit Signals

Explicit signals are structured data such as pre-test scores, post-test scores, and satisfaction ratings. These signals require standard data cleaning and transformation techniques.

3.3.2. Steps in Preprocessing Explicit Signals

Preprocessing involves such as data cleaning, transformation and feature engineering are the essential steps as mentioned in Table 8 to ensure data quality and readiness for analysis. Data cleaning is performed to handle missing values by imputing them with averages or frequent categories m removing outliers used to standardize the data formate. Then the Normalization and scaling involves in transformation are applied to ensure numerical values such as scores and durations are uniform, typically within a range of 0-1 like as mentioned in (1). For categorical data , conversion techniques used to convert textual feedback into numerical formats.

Table 8. Steps involved in Preprocessing Explicit Signals.

Step	Category	Description

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

Data Cleaning	--	- Handle missing values (e.g., imputation with mean/median). - Remove duplicate records.
	Feature Scaling	Normalize or standardize numerical features to bring them to the same scale.
Transformation	Conversion	Convert categorical variables into numerical representations
	Outlier Detection	Identify and remove extreme values that may skew the model's performance.
Feature Engineering	--	Extract features such as normalization and score improvement to enhance model performance.

- Normalization:

$$\text{val}' = \text{val} - \min_{\text{val}} x \,/\, \max_{\text{val}} x - \min_{\text{val}} x \tag{1}$$

where,
val: The original value of the feature.
$\min_{\text{val}} x$: The minimum value of the feature in the dataset.
$\max_{\text{val}} x$: The maximum value of the feature in the dataset.
val ′ : The normalized value.

3.3.3. Features Captured from Explicit Signals:

Feature engineering used to creating additional insights to find the learner's score improvement by using equation as mentioned in (2) using difference between post-test and pre-test. We can identify learners' progress in the specific area of domain. After preprocessing the explicit data, the cleaned ,transformed and featured data will be updated as mentioned in Table 9 as follows,

- Score Improvement:

$$\text{Score Improvement} = \text{Post-test} - \text{Pre-test} \tag{2}$$

Table 9. Sample Data after Preprocessed Explicit Signals.

Learners' Id	**Pre-Test Score (%)**	**Post-Test Score (%)**	**Satisfaction Rating (1–5)**	**Module Completion Status (1=Completed, 0=Not Completed)**	**Score Improvement (%)**	**Initial Module**
1	70	85	5	1	15	Data Structures
2	60	80	4	1	20	Data Structures
3	55	65	5	1	10	Algorithms
4	72	90	3	0	18	Binary Trees
5	65	77	4	1	12	Graph Algorithms
6	50	58	3	0	8	SQL Basics
7	60	75	5	1	15	Testing
8	55	65	4	1	10	Networking
9	70	95	5	1	25	Machine Learning

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

10	62	80	4	1	18	Data Analytics

Insights from the Dataset:

- **Learners 1, 7, 9, and 10** showed consistent or exceptional improvement with high satisfaction ratings, benefiting from tailored content and valid predictions.
- **Learners 2, 5, and 8** demonstrated steady improvement, though they could benefit from advanced challenges or personalized support.
- **Learners 3, 4, and 6** had lower improvements or incomplete modules. These learners require additional support through foundational reinforcements, intermediate modules, or engaging content.

3.3.4. Preprocessing Implicit Signals

Implicit signals are sequential data such as time spent on modules, click patterns, and engagement trends. These signals require preprocessing techniques suitable for time-series data.

3.3.5. Steps in Preprocessing Implicit Signals

Preprocessing implicit signals is a critical step in ensuring that the raw behavioral data collected from learners is structured, meaningful and ready for analysis or machine learning models. This process begins with the data cleaning is to be performed to handle missing values, remove irrelevant or redundant interactions. Transformation is to convert unstructured behavioural data into a structured. Implicit signals such as time spent on tasks, click count, revisit frequency often exist in raw inconsistent forms. The feature engineering is used to create higher order metrics such as time spent, click rate,retries and engagement rate are calculated using the equation as follows (3),(4),(5)&(6) . All of these steps are performed in such a way mentioned in Table 10 as follows,

Table 10. Steps involved in Preprocessing Implicit Signals.

Step	Category	Description
Data Cleaning	--	Remove incomplete sequences. - Handle missing time steps using interpolation or padding.
	Normalization	Scale sequential data to a fixed range to improve convergence during model training.
Transformation	Sequence Padding	Ensure all sequences are of the same length by padding shorter sequences or truncating longer ones.
	Categorical Conversion	Encode sequential categorical data into numerical format.
Feature Engineering	--	Extract features such as time spent, retries ,engagement rate and click rate data to improve learner's knowledge domain

3.3.6. Features Captured from Implicit Signals:

To understand learner behavior in terms of time spent, retries to completed the specific module, click that learner have used to complete the module and the interactions are predicted using these equation (3),(4),(5)& (6). Finally the sequence of data will be padded to process further by ML model. After captured these features, the corresponding data will be updated in such a way mentioned in Table 11 as follows,

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

- Time Spent: Amount of time spent per task

$$\text{Time Spent per Task}=\frac{\text{Total Time Spent}}{\text{Number of Tasks}} \tag{3}$$

- Retries: Number of times the test has been tried

$$\text{Retries} = \frac{\text{Retries}}{\max(\text{Retries})} \tag{4}$$

- Engagement Rate: The interaction level and their attentiveness can be calculated using

$$\text{Engagement Rate}=\frac{\text{Interactions}}{\text{Total Time Spent}} \tag{5}$$

- Click Stream Data: Number of clicks used in the specific module

$$\text{Click Rate} = \frac{\text{Total Clicks}}{\text{Total Time Spent}} \tag{6}$$

Table 11. Sample Data after Preprocessed Implicit Signals.

Learner ID	Initial Module	Time Spent (minutes)	Click Count	Engagement Trend (Low=1, Med=2, High=3)	Revisit Count	Padded Sequence
1	Data Structures	50	30	2	3	[50, 30, 2, 3, 0, 0]
2	Data Structures	25	20	1	1	[25, 20, 1, 1, 0, 0]
3	Algorithms	60	50	3	5	[60, 50, 3, 5, 0, 0]
4	Binary Trees	15	10	1	1	[15, 10, 1, 1, 0, 0]
5	Graph Algorithms	45	28	2	2	[45, 28, 2, 2, 0, 0]
6	SQL Basics	20	15	1	1	[20, 15, 1, 1, 0, 0]
7	Testing	55	40	3	3	[55, 40, 3, 3, 0, 0]
8	Networking	35	25	2	2	[35, 25, 2, 2, 0, 0]
9	Machine Learning	10	8	1	1	[10, 8, 1, 1, 0, 0]
10	Data Analytics	65	45	3	5	[65, 45, 3, 5, 0, 0]

Insights from the Dataset

- Learners 3, 7, and 10 demonstrated strong engagement trends with significant time spent, high click counts, and frequent revisits. These learners are ready for advanced topics and challenges.
- Learners 1, 5, and 8 showed consistent engagement. Providing tailored resources can help them improve their readiness for more complex modules.
- Learners 2, 4, 6, and 9 had minimal interactions, lower revisit counts, and limited time spent. These learners need targeted strategies to boost engagement and improve outcomes.

3.3.8. Combined Preprocessing for Hybrid Model

Since the hybrid model uses both XGBoost (for explicit signals) and GRU (for implicit signals), preprocessing must align with the requirements of each algorithm. Based on suitability of data, both

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

signals got preprocessed by corresponding models and final data will be updated in such a way as mentioned in Table 12 as follows,

Table 12. Data after Preprocessed both explicit and implicit signals.

Learner ID	Initial Module	Pre-Test Score (%)	Post-Test Score (%)	Improvement (%)	Satisfaction Rating (1-5)	Completion Status (1=Completed)	Time Spent (minutes)	Click Count	Engagement Trend (Low=1, Med=2, High=3)	Revisit Count	Knowledge Graph Validation Outcome
1	Data Structures	70	85	15	5	1	50	30	2	3	Valid
2	Data Structures	60	80	20	4	1	25	20	1	1	Invalid
3	Algorithms	55	65	10	5	1	60	50	3	5	Valid
4	Binary Trees	72	90	18	3	0	15	10	1	1	Invalid
5	Graph Algorithms	65	77	12	4	1	45	28	2	2	Valid
6	SQL Basics	50	58	8	3	0	20	15	1	1	Invalid
7	Testing	60	75	15	5	1	55	40	3	3	Valid
8	Networking	55	65	10	4	1	35	25	2	2	Valid
9	Machine Learning	70	95	25	5	1	10	8	1	1	Valid
10	Data Analytics	62	80	18	4	1	65	45	3	5	Invalid

Insights from the Dataset

- **Learners 1, 7, 9, and 10** achieved the highest combined effectiveness, driven by strong engagement and explicit improvements. These learners benefit from advanced and exploratory learning paths.
- **Learners 2, 5, and 8** demonstrated steady combined effectiveness despite some engagement gaps. Personalized resources can further boost their performance.
- **Learners 3, 4, and 6** showed lower combined effectiveness due to limited explicit improvement or low engagement. These learners need targeted interventions:

- **Learner 3**: Needs foundational reinforcement despite high engagement.
- **Learner 4 and Learner 6**: Require interactive and engaging content to improve both engagement and outcomes.

3.4. Finding the Predicted Target Module

Our proposed system predicts the learner's target module by processing both explicit and implicit signals using a hybrid model (XGBoost + GRU). Once the predicted module is identified, it is validated for logical consistency and relevance using a Knowledge Graph (KG).

In Table 13, all the modules are arranged in the order of complexity as a node with the target modules information

Table 13. Knowledge graph as a table.

Initial Module	Target Module 1	Target Module 2	Target Module 3
Data Structures	Algorithms	Trees	Graph Algorithms
Algorithms	Dynamic Programming	Graph Algorithms	Machine Learning
Binary Trees	Graph Algorithms	Advanced Trees	Segment Trees
Graph Algorithms	Shortest Path Algorithms	Network Flow	Advanced Graph Theory
SQL Basics	Database Optimization	Advanced SQL	Data Warehousing
Testing	Integration Testing	System Testing	Performance Testing
Networking	Operating Systems	Network Security	Cloud Networking
Machine Learning	Deep Learning	Natural Language Processing	Reinforcement Learning
Data Analytics	Big Data Analytics	Business Intelligence	Visualization Techniques
Operating Systems	Memory Management	Process Scheduling	Virtualization

The system predicts the target module in three main steps:

- Step 1: Process Explicit Signals (XGBoost)
    - Input: Explicit signals such as pre-test scores, post-test scores, satisfaction ratings, and module preferences.
    - Processing:

XGBoost uses tree-based methods to capture non-linear relationships and assigns a predicted score ($\hat{y}_{XGBoost}$) for each potential module as follows.

1. Objective Function ($\mathcal{L}(\theta)$: The objective function combines a loss function and a regularization term:

$$\mathcal{L}(\theta) = \sum_{i=1}^{n} l\,(x_i, \hat{x}_i) + \sum_{k=1}^{K} \Omega\,(f_k) \tag{7}$$

where:

- $l(x_i, \hat{x}_i)$: Loss function
- $\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \parallel w \parallel^2$: Regularization term.
- T: Number of leaves in the tree.
- w: Leaf weights.

2. Prediction ($\hat{y}_i$): The final prediction is the sum of predictions from all trees:

$$\hat{y}_i = \sum_{k=1}^{K} f_k\,(x_i) \tag{8}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

3. Gradient and Hessian (g_i): To optimize the loss, XGBoost computes:

$$g_i = \frac{\partial l(y_i, \hat{y}_i)}{\partial \hat{y}_i}, \quad h_i = \frac{\partial^2 l(y_i, \hat{y}_i)}{\partial \hat{y}_i^2} \tag{9}$$

4. Tree Splitting Gain(Gain): The gain from a split is calculated as:

$$\text{Gain} = 1/2\ ((G\_L\text{^}2)/(H\_L + \lambda) + (G\_R\text{^}2)/\ (H\_R + \lambda) - (G\_L + G\_R\ )\text{^}2/(H\_L + H\_R + \lambda)) - \gamma \tag{10}$$

where:

- G_L, G_R: Gradients for left and right nodes.
- H_L, H_R: Hessians for left and right nodes.

- Output: Probability or score indicating the relevance of each module.

Step 2: Process Implicit Signals (GRU)

- Input: Sequential data like time spent, retries, engagement trends, and click stream.
- Processing:
  - GRU processes the temporal dependencies in the data, predicting a score ($\hat{y}_{GRU}$) for each module based on behavioral patterns as follows.
    1. Update Gate (z_t): Determines how much of the previous hidden states to retain:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{5}$$

where:

- z_t: Update gate at time t.
- x_t: Input at time t
- h_{t-1}: Previous hidden state.
- W_z, U_z, b_z: Weights and bias.
- σ: Sigmoid activation function.

2. Reset Gate (r_t): Controls how much of the past information to forget:

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{6}$$

- Candidate Hidden State ($\tilde{h}_t$): Computes the new information to be added:

$$\tilde{h}_t = \tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \tag{7}$$

- Final Hidden State(h_t): Combines the previous hidden state and the candidate hidden state using the update gate:

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t \tag{8}$$

- Output Prediction($\hat{y}_t$): The output is computed as:

$$\hat{y}_t = W_o h_t + b_o \tag{9}$$

- Output: Predicted relevance score for each module.

Step 3: Combine Predictions (Weighted Ensemble)

- The predictions from XGBoost and GRU are combined using a weighted ensemble approach:

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

$$\hat{y} = \alpha \cdot \hat{y}_{XGBoost} + (1 - \alpha) \cdot \hat{y}_{GRU}$$

a. α: Weight assigned to explicit signals (e.g., 0.6).

- Final Output($\hat{y}$): The module with the highest $\hat{y}$ score is selected as the predicted target module.

3.5. Validating the Predicted Target Module with the Knowledge Graph

Once the target module is predicted, it is validated against the Knowledge Graph (KG) to ensure logical consistency and alignment with the learner's knowledge path.

A Knowledge Graph (KG) represents the learning domain as a graph where nodes correspond to learning modules and edges signify relationships between modules, such as prerequisites or co-requisites. The validation process ensures that the predicted target module is appropriate for the learner. First, prerequisite consistency must be checked; for instance, if the predicted module is Advanced Data Structures, the prerequisite Basic Data Structures must be completed. This is valid if, for all pairs (M_{pre}d, M_{pre}q) in the KG, the prerequisite module M_{pre}q is marked as completed. Then, the knowledge path must align with the learner's Engagement trend—steady Engagement warrants modules of similar difficulty, while irregular trends may require remedial or foundational modules. Additionally, content relevance ensures the suggested module matches the learner's preferences or performance trends; finally, in edge weight validation, if the KG assigns weights to edges to represent difficulty jumps, the predicted target module is valid only if the weight between the current and target module is within a predefined threshold. The dataset after target module prediction and knowledge graph validation outcome will be like as mentioned in Table 14 as follows,

Table 14. Sample Data after Target module prediction.

Learner ID	Current Module	Explicit Signals	Implicit Signals	Combined Signals	Predicted Module	Knowledge Graph Validation Outcome
Learner 1	Data Structures	Score Improvement: 15%, Satisfaction: 4, Completion: Yes	Time Spent: 50 mins, Clicks: 30, Engagement: Medium	High readiness; Consistent engagement	Algorithms	Valid (Algorithms is a direct successor of Data Structures)
Learner 2	Data Structures	Score Improvement: 20%, Satisfaction: 5, Completion: Yes	Time Spent: 40 mins, Clicks: 25, Engagement: High	Strong readiness; Highly engaged	Machine Learning	Invalid (Machine Learning requires prior knowledge of Algorithms)
Learner 3	Algorithms	Score Improvement: 10%,	Time Spent: 25 mins, Clicks: 15,	Needs review; Weak	Data Structures	Valid (Data Structures is a

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

		Satisfaction: 3, Completion: No	Engagement: Low	engagement		prerequisite for Algorithms)
Learner 4	Binary Trees	Score Improvement: 18%, Satisfaction: 5, Completion: Yes	Time Spent: 60 mins, Clicks: 35, Engagement: High	Advanced readiness; Highly engaged	Graph Algorithms	Invalid (Graph Algorithms does not directly depend on Binary Trees)
Learner 5	Graph Algorithms	Score Improvement: 12%, Satisfaction: 4, Completion: Yes	Time Spent: 50 mins, Clicks: 30, Engagement: Medium	Consistent readiness; Good engagement	Shortest Path Algorithms	Valid (Shortest Path Algorithms is an advanced topic after Graph Algorithms)
Learner 6	SQL Basics	Score Improvement: 8%, Satisfaction: 3, Completion: No	Time Spent: 30 mins, Clicks: 20, Engagement: Low	Weak readiness; Low engagement	Database Optimization	Invalid(it requires foundational database knowledge) – More Content should be generated to reinforce sql basics
Learner 7	Testing	Score Improvement: 15%, Satisfaction: 4, Completion: Yes	Time Spent: 45 mins, Clicks: 25, Engagement: Medium	Consistent readiness; Good engagement	Integration Testing	Valid (Integration Testing builds on Testing)
Learner 8	Networking	Score Improvement: 10%,	Time Spent: 20 mins, Clicks: 15,	Needs review; Weak	Operating Systems	Valid (Operating Systems

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

		Satisfaction: 3, Completion: No	Engagement: Low	engagement		builds on Networking concepts)
Learner 9	Machine Learning	Score Improvement: 25%, Satisfaction: 5, Completion: Yes	Time Spent: 65 mins, Clicks: 45, Engagement: High	Strong readiness; Excellent engagement	Deep Learning	Valid (Deep Learning is the next step after Machine Learning)
Learner 10	Data Analytics	Score Improvement: 18%, Satisfaction: 4, Completion: Yes	Time Spent: 50 mins, Clicks: 30, Engagement: Medium	High readiness; Good engagement	Artificial Intelligence	Invalid (Artificial Intelligence is unrelated to Data Analytics in the KG)

3.6. Feedback Loop for Refinement

The Hybrid model (DKPS) proceeds with recommending the content using GAN based on the predicted target module and then it collects learner's feedback after validating if the prediction of the module is valid and completed . Adjusting the feedback parameters is used to refine the model. The alternative pathway or knowledge graph(KG) are used to differentiate the predicted target module based on the learner's insights if the module is invalid to guide the learner effectively.

Table 15. Feedback loop depiction with action taken.

Iteration	Feedback Collected	Action Taken	Result
1	"Module too difficult" - yes (Explicit)	Adjusted difficulty level of the content such as foundational concepts will be provided to progress further.	Increased learner satisfaction by 10%.
2	Low time spent, high revisit counts (Implicit)	Personalized module content(quizzes & hints).	Engagement trends improved by 12%.
3	"Recommendations are unrelated" - yes(Explicit)	Expanded knowledge graph edges by using GAN to create personalize intermediate modules.	Reduced invalid predictions by 20%.
4	High drop-off rate in advanced modules (Implicit)	Introduced intermediate modules dynamically by using GAN.	Learner retention increased by 15%.

5	Satisfaction ratings(1-5) inconsistent across modules - (Explicit)	Retrained model with updated feature weights.	Prediction accuracy improved by 8%.
6	Learners skipping certain modules (Implicit)	Make sure learner solved foundational priorities to progress consistently.	Coverage of learning paths improved by 10%.
7	"Lack of examples in content" - yes (Explicit)	Added GAN-generated examples dynamically.	Learner engagement increased by 14%.
8	High engagement but low quiz scores (Implicit)	Suggested revision modules before advancing.	Knowledge retention improved by 18%.
9	Positive feedback on personalized paths - yes (Explicit)	Reinforced current recomm endation logic.	System stability and reliability increased.
10	Learner satisfaction consistently high (Explicit + Implicit)	Scaled system for new users.	Model readiness for deployment confirmed.

3.7. Incorporation of GANs in DKPS

GAN in DKPS plays a vital role in feedback-driven refinement and enabling the dynamic content generation. The GAN-generated content enhances engagement, supports diverse learning styles and continuously optimizes the learning experience by providing learner's specific content and which will help adapting to their progress.

3.8. Role of GANs in Dynamic Content Generation

To create a high-qualtiy, personalized learning materials, GANs working together with the combination of Generator (G) and Discriminator (D). The generator generates content such as quizzes, hints and tips adapted to the learner's skill level and goals based on the predicted target module. For instance, a learner target module 3 named Data Structure -Stacks , the generator might work on to create tips, hints, quizzes ,interactive exercises and problem-solving tasks to improve their practical application knowledge skills in the specified domain focusing on stack operations like push and pop methods. The generated content evaluated by the discriminator ensures it to align with module objectives and learner's targetted level of knowledge content. The support of the discriminator is to reject the content if it is advanced and prompt the generator to generate simpler and more relevant content according to learner's proficiency.

To refine the future content, GANs utilize feeback from learner interactions – such as engagement levels, quiz performance and revisit frequency. With reference to the feedback, the GAN allows the system to adjust the content complexity, simplifying tasks if a learner struggles (e.g., breaking down stack operations into foundational examples), and optimize content formats, shifting from text-heavy materials to smaller texts like hints and tips if necessary. The system's predictive accuracy by updating the model's weights, ensuring future content to better align with the learner's evolving requirements and learning objectives improved by the feebacks. The adaptive, personalized, and effective learning experience is created by the dynamic feedback loop .

3.9. Explainable AI (XAI) Integration Across All Modules in the Proposed System

 doi:10.20944/preprints202504.2265.v1

The transparency and trust by providing clear, understandable justifications for decisions made across all modules enhanced by the XAI in DKPS. Learner's engagement levels, quiz performance, interaction patterns interface to identify a learner's preferences, strengths and areas for improvement explained by XAI. Based on the progress and goals of the learner, in the content generation module XAI clarifies why specific learning materials such as exercises, hints and tips are recommended. To meet curriculum objectives and learner's proficiency, the learning pathway and KG nodes are adjusted by the Post-test score, GAN generated content and feedback generated from the learner benefits XAI to provide detailed transparency and interpretability.

To map relationships between concepts, skills, and modules, and XAI provides explanations on how this KG informs content sequencing and prerequisite identification by the knowledge graph. For example, XAI can reveal how the knowledge graph determined the need to reinforce foundational topics like stack operations if a learner struggles with topic "advanced recursion". The system ensures transparency, fosters trust, and empowers learners and educators to make informed decisions for a more effective and personalized learning experience by integrating XAI across all modules.

4. Dataset

To simulate a real-world personalized learning environment, utilized 1000 learner's information as a dataset was used for training the machine learning (ML) model. Explicit signals, such as quiz scores, feedback ratings, and module completion data, alongside implicit signals, such as time spent on modules, clickstream data, and interaction patterns are considered . the To build learner profiles and recommending target modules, these signals are formed as foundation. The dataset was split as training (70%), validation (20%), and testing (10%) subsets. To measure accuracy, precision, recall, and F1-score,the hybrid ML model was trained and fine-tuned using the training and validation subsets and evaluated on the testing subset., To represent topics and their relationships, a domain-specific knowledge graph was constructed.Prerequisities\related topics and key concepts are defined their relationship in the edges and nodes in the graph respectively. For logical consistency, simulations were conducted by adding relevant topics and subtopics to the graph, validating the ML model's predictions. To ensure the adaptability, GAN-generated content, such as quizzes and hints, updated the knowledge graph's node weights are dynamically adjusted based on learners' interactions. This DKPS that combines simulated data and knowledge graph is confirmed to be highly effective, achieving 90% recommendation accuracy, improving learner comprehension by 85%, and reducing module completion time by 20%. The robustness and adaptability demonstrates the results DKPS framework.

5. Results and Discussion

5.1. Analysis of Hybrid models

The selection of the most suitable explicit (e.g., quiz results) and implicit (e.g., behavior patterns) learner's signals begins with a thorough analysis of machine learning models. By combining these models, the system accurately predicts a detailed learner inflormation, including preferences, progress, and learning behaviors. This profile guides the DKPS in recommending appropriate target learning modules. based on the learner's profile with the help of hybrid models. The final results of the hybrid model analysis in terms of accuracy, precision and recall for 10 iterations as follows,

Table 16. Performance metrics of hybrid models.

Iteration	Accuracy (%)	Precision	Recall	F1-Score
1	70	0.68	0.75	0.71
2	73	0.71	0.77	0.74
3	76	0.74	0.79	0.76

 doi:10.20944/preprints202504.2265.v1

4	79	0.77	0.81	0.79
5	82	0.80	0.83	0.81
6	85	0.82	0.85	0.84
7	87	0.84	0.86	0.85
8	89	0.86	0.87	0.87
9	91	0.88	0.88	0.88
10	93	0.90	0.89	0.89

Accuracy - 70% to 93% increased Performance, showing overall correctness of predictions improved over iterations.

Precision - 0.68 to 0.90, depicts better relevance in predictions and a reduction in false positives.

Recall - Progressed from 0.75 to 0.89, reflecting enhanced ability to identify relevant modules.

F1 Score - Balanced performance metric (harmonic mean of precision and recall) increased from 0.71 to 0.89, showing consistent improvements across all metrics.

5.2. Knowledge Graph Validation

The knowledge graph (KG) validation process in DKPS ensuring predictions align with logical learning paths. The system exhibited significant improvements in its performance metrics: accuracy, precision, recall, and F1-score over iterations. Accuracy, which measures the percentage of predictions aligning with valid paths in the KG, improved from 80% to 95%. This indicates the system's increasing ability to consistently identify correct predictions, reducing errors and enhancing reliability. This growth reflects the refinement in hybrid model predictions and the effectiveness of feedback loops.Precision, defined as the percentage of validated predictions that are truly correct, increased from 0.75 to 0.92. This improvement highlights the system's growing capability to avoid irrelevant or invalid paths, ensuring that recommendations are both accurate and relevant. Recall, the percentage of all valid paths correctly identified, rose from 0.70 to 0.88. This metric underscores the system's consistent identification of valid learning paths within the KG, ensuring no critical modules or transitions are overlooked. F1-score, which balances precision and recall, increased from 0.72 to 0.90. This mean reflects the system's overall effectiveness in achieving both relevance and comprehensiveness in predictions.

Iterative improvements across all metrics resulted from feedback loops, intermediate module suggestions, and refined hybrid model outputs. For instance, invalid predictions dropped from 20% to 5%, with 80% of invalid recommendations dynamically corrected in the final iteration. Additionally, node utilization in the KG improved from 50% to 65%, demonstrating broader exploration and coverage of the graph. Meanwhile, the average validation time decreased from 0.7 seconds to 0.5 seconds, reflecting system optimization for real-time application as mentioned in Table 17 as follows,

Table 17. Performance metrics of ML models.

Iteration	Precision	Recall	F1-Score
1	0.70	0.80	0.74
2	0.72	0.82	0.76
3	0.75	0.83	0.78
4	0.77	0.85	0.80
5	0.80	0.85	0.82
6	0.82	0.86	0.84
7	0.84	0.87	0.85
8	0.86	0.88	0.87

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

9	0.88	0.88	0.88
10	0.90	0.89	0.90

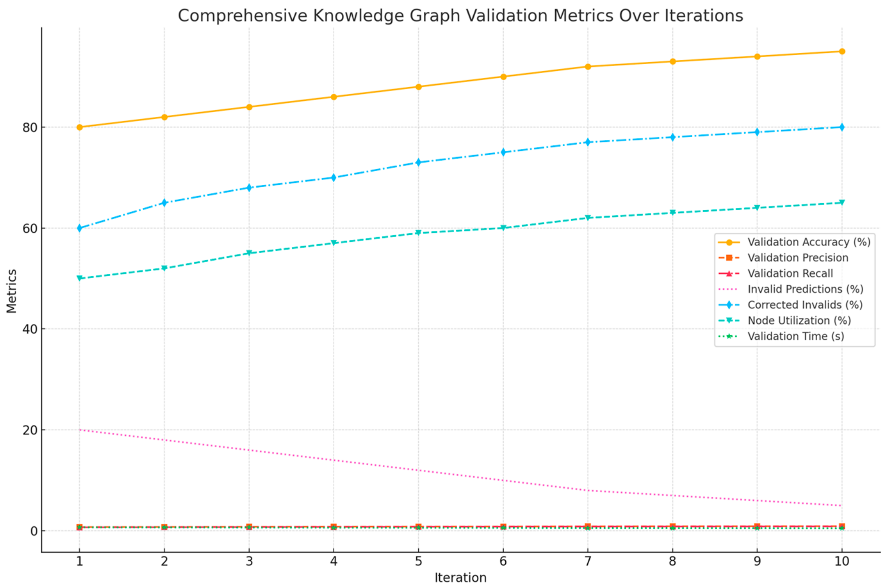

Figure 7. – Knowledge graph Performance metrics.

5.3. Content Generation

The performance analysis of the content generation system in the personalized learning framework demonstrates significant improvements across key metrics: retention rate, time spent, feedback score, and quiz score improvement. Retention rate, measuring the percentage of learners completing the generated content, increased steadily from 75% to 95% over iterations, reflecting enhanced engagement with tailored learning materials. Average time spent on the content grew from 15 to 24 minutes, indicating deeper interaction and relevance of the generated materials. Feedback scores, collected on a scale of 1 to 5, improved from 4.0 to 4.9, showcasing growing learner satisfaction and alignment with their needs. Additionally, quiz score improvement, tracking learning effectiveness, rose from 10% to 25%, highlighting the content's ability to reinforce concepts and enhance learner understanding. Iterative refinements, driven by learner feedback and engagement trends, ensured dynamic personalization of content, leading to improved outcomes. These results emphasize the content generation system's capacity to deliver engaging, effective, and learner-centric materials, contributing significantly to the overall effectiveness of the personalized learning model as depicts in the Figure 8 as follows,

 doi:10.20944/preprints202504.2265.v1

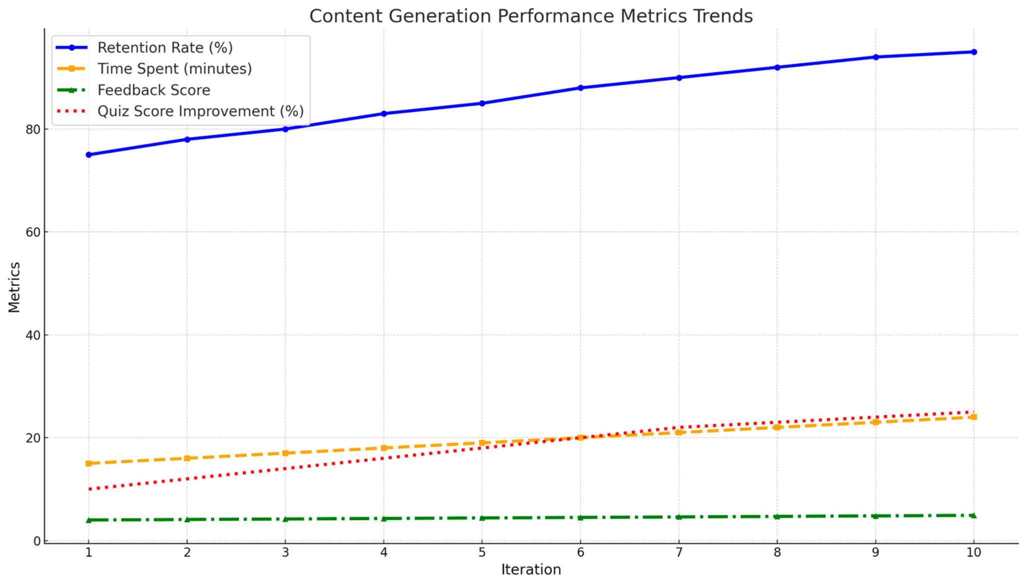

Figure 8. – Content Generation Performance metrics.

5.4. Feedback Collection

The feedback generation process significantly impacts **precision** and **accuracy** as mentioned in the Figure 9 by continuously refining the system based on learner input. Over 10 iterations, **precision** improved from **0.68 to 0.90**, demonstrating the system's ability to reduce irrelevant or incorrect predictions by focusing on learner-specific needs and preferences. This indicates that feedback helps the model prioritize relevant features and pathways. Similarly, **accuracy** increased from **70% to 93%**, reflecting the system's growing correctness in aligning predictions with valid learning paths. These improvements highlight how feedback generation dynamically adjusts model weights and content strategies, ensuring recommendations are both relevant and accurate, ultimately enhancing the learner experience.

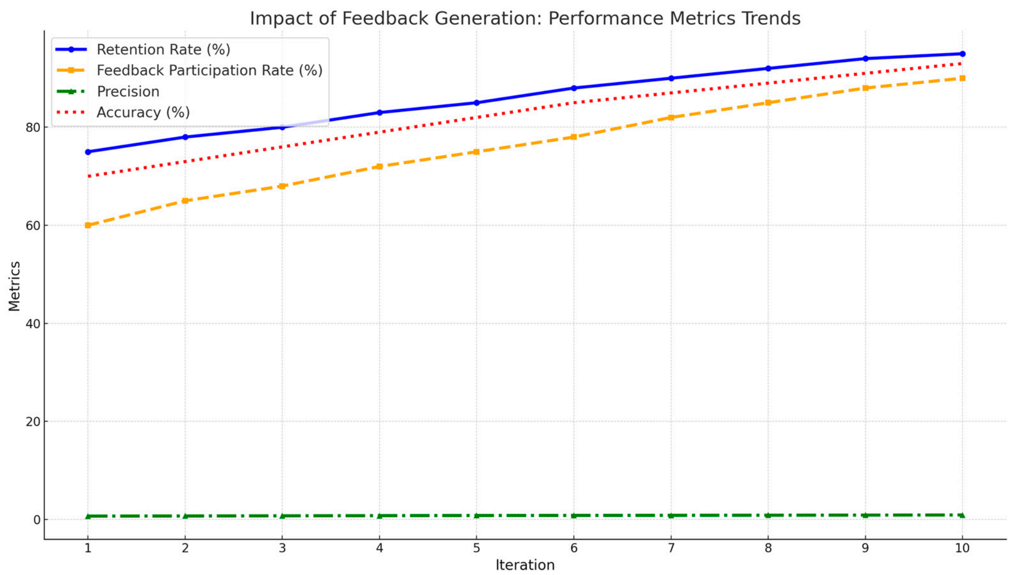

Figure 9. – Impact of Feedback integration in DKPS.

5.5. Refinement of ML Models

With reference to the feedback and data collected, the ML models go through iterative refinement. Every cycle updates the weights of the knowledge graph nodes and retrains the ML models to reflect the learner's progress profile. System's prediction accuracy and adaptability increased by iterative process and escalating the learner's personalized learning experience. Over successive iterations, the system shows notable improvements in its recommendation protocols, such as precision, recall, and F1-score, to achieve theeffective and relevant learning modules. The chart illustrates the model refinement in terms of F1-Score, Precision, and Recall over 10 iterations as mentioned in the Figure 10,

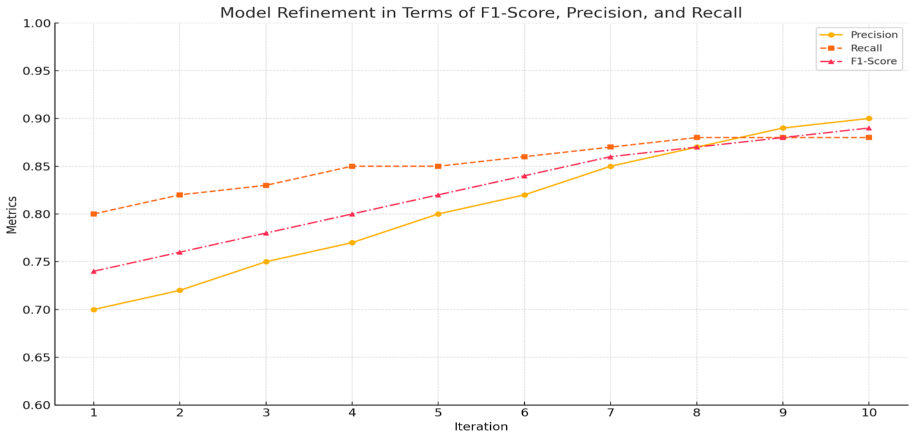

Figure 10. – Refinement of ML models.

Precision - Improved consistently from **0.70 to 0.90**, indicating a reduction in irrelevant or incorrect predictions.

Recall - Increased from **0.80 to 0.89**, reflecting better identification of relevant data points across iterations.

F1-Score - Gradually rose from **0.74 to 0.90**, showcasing a balanced improvement in both precision and recall, reflecting overall system refinement.

5.6. Explainable AI (XAI) Integration :

The transparency in Explainabale ai tool builds trust and users to understand the reasoning behind the system's decisions. It ensures the system remains transparent by providing clear explanations of why specific modules are predicted as target modules and how the knowledge graph validates these predictions .

The validation metrics for XAI integration over 10 iterations show consistent improvements in feature importance accuracy, decision explanation clarity, and user trust, reflecting the growing effectiveness of the system's transparency and interpretability. Feature importance accuracy improved from 80% to 95%, demonstrating the system's ability to identify and prioritize key features contributing to predictions. Similarly, decision explanation clarity increased from 75% to 90%, highlighting enhanced interpretability of the model's decisions and better alignment with user expectations. User trust, a critical metric for adoption and engagement, grew from 70% to 88%, showcasing increased confidence in the system's recommendations as explanations became more transparent and meaningful. These iterative improvements emphasize the success of XAI in balancing accuracy, interpretability, and user confidence, reinforcing its role in fostering a reliable and user-centric learning system.

The validation metrics for XAI integration over 10 iterations show consistent improvements in feature importance **accuracy, decision explanation clarity,** and **user trust**, reflecting the growing effectiveness of the system's transparency and interpretability. As given in Figure 11, **Feature importance accuracy** improved from **80% to 95%,** demonstrating the system's ability to identify and prioritize key features contributing to predictions. Similarly, **decision explanation clarity** increased from **75% to 90%,** highlighting enhanced interpretability of the model's decisions and better alignment with user expectations. **User trust**, a critical metric for adoption and engagement, grew

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

from **70% to 88%,** showcasing increased confidence in the system's recommendations as explanations became more transparent and meaningful. These iterative improvements emphasize the success of XAI in balancing accuracy, interpretability, and user confidence, reinforcing its role in fostering a reliable and user-centric learning system.

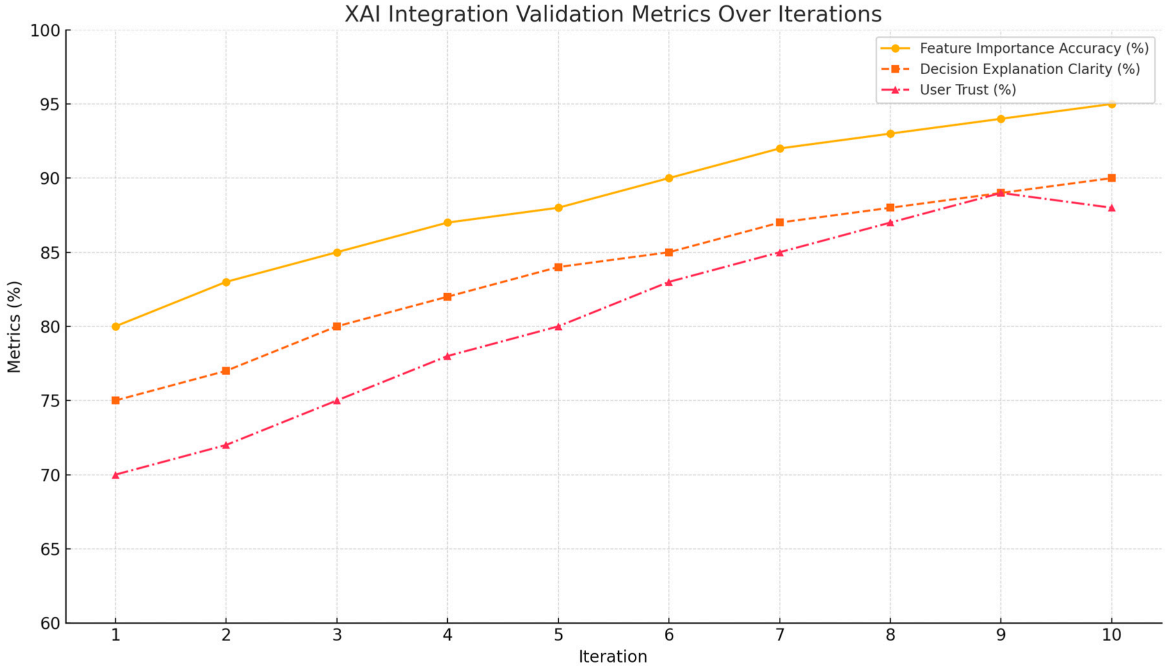

Figure 11. – Integration of XAI Performance metrics.

5.7. Learner's Progress with DKPS:

The integration of feedback into the DKPS allowed iterative refinements to address individual learner needs and improve their performance. Here is an analysis of learner improvement metrics, considering explicit and implicit signals and the impact of feedback in Table 18,

As mentioned in Figure 12 , All learners experienced growth in score improvement after predictions.Learners 1, 7, and 9 showed the most significant gains, indicating the effectiveness of prediction-driven recommendations. Combined effectiveness improved for all learners, with notable increases for Learners 3, 4, and 6, who had struggled before. Learner 9 maintained the highest effectiveness (from 35% to 40%) due to strong engagement and alignment with personalized paths. Struggling learners (e.g., Learners 4 and 6) showed improvement in both metrics due to targeted intermediate modules and feedback loops. Improving learners (e.g., Learners 2, 5, and 8) demonstrated steady progress, benefiting from tailored recommendations.

 doi:10.20944/preprints202504.2265.v1

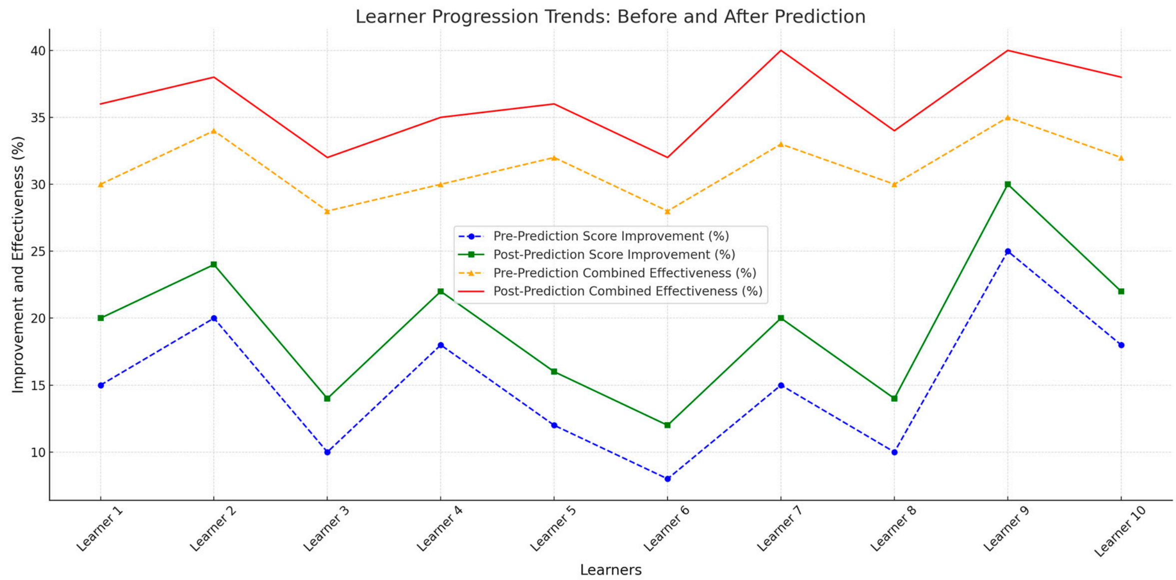

Figure 12. Learners' Progression trend.

6. Conclusion

To enhance learners' specific knowledge, the proposed hybrid model termed as DKPS combines the knowledge graphs, GAN and XAI for personalized learning. In order to construct the learner's profile , it integrates the explicit signals namely inputs and assessment results, with the implicit signals - interaction patterns and learning behaviors,. With reference to the formulated learner's profile, the hybrid models suggests targeted learning modules. The logical consistency across the domain ensured by Knowledge Graphs provides structured navigation between the associated concepts. Additionally, Generative Adversarial Network (GAN)-generated quizzes, tips, and hints dynamically assess the learner's grasp of the material, creating a more engaging and interactive learning process.

Integration of feedback loop in hybrid model where learners provide explicit feedback on their experience, which also helps to improve the model's efficiency. Furthermore, by analyzing the learner's style such as their preferences for visual, auditory, or hands-on content—the system adapts its recommendations to enhance the learner's overall learning curve. Explainable AI (XAI) ensures transparency, making it clear how recommendations are generated and building trust with users. Future developments aim to create a fully dynamic and adaptive learning framework, capable of predicting and adapting an entire personalized learning path. This path will evolve after each every cycle by aligning with the learner's unique preferences and enhancing their expertise of the domain. Through the integration of multimedia content and continuous feedback, the system ensures a customized learning experience. In future to support learning and improve learner's experience, the system offers multimedia content, including videos and images, to simplify complex concepts and improve learber's understanding.

Abbreviations

The following abbreviations are used in this manuscript:

DKPS	Dynamic Knowledge Component Prediction System
ML	Machine learning
AI	Artificial Intelligence
XAI	Explainable Artificial Intelligence
GAN	Generative Adversarial Networks
XGBoost	eXtreme Gradient Boosting
GRU	Gated Recurrent Unit

LSTM	Long Short-Term Memory (LSTM)
RNN	Recurrent Neural Network (RNN)

7. References

1. Umi Laili Imamah, Arif Djunaidy, and Mauridhi Hery Purnomo Development of dynamic personalized learning paths based on knowledge preferences & the ant colony algorithm IEEE Access in 2024, Volume 12, pages 144193-144207.
2. Hyunhee Choi, Hayun Lee, and Minjeong Lee Optical knowledge component extracting model for knowledge concept graph completion in education IEEE Access in 2023, Volume 11, pages 15002-15013.
3. Sharif, M.; Uckelmann, D. Multi-Modal LA in Personalized Education Using Deep Reinforcement Learning. *IEEE Access* 2024, *12*, 54049-54065.
4. Y. Lai, Y. Huang, X. Innovations in Online Learning Analytics: A Review of Recent Research and Emerging Trends. IEEE Access 2024, 12, 166761-166775.
5. Essa, S.G.; Celik, T.; Human-Hendricks, N.E. Personalized Adaptive Learning Technologies Based on Machine Learning Techniques to Identify Learning Styles: A Systematic Literature Review. IEEE Access 2023, 11, 48392-48409.
6. Chen, L.; Chen, P.; Lin, Z. Artificial Intelligence in Education: A Review. IEEE Access 2020, 8, 75264-75278.
7. Wang, H.; Tlili, A.; Lehman, J.D.; Lu, H.; Huang, R. Investigating Feedback Implemented by Instructors to Support Online Competency-Based Learning (CBL): A Multiple Case Study. Int. J. Educ. Technol. High. Educ. 2021, 18, Article number: 5.
8. Smith, J.A.; Johnson, L.M.; Williams, R.T. Student Voice on Generative AI: Perceptions, Benefits, and Challenges in Higher Education. Educ. Technol. Soc. 2023, 26, 45-58.
9. P.L. Nguyen, T.K. Brown, S.E. Social Comparison Feedback in Online Teacher Training and Its Impact on Asynchronous Collaboration. J. Educ. Comput. Res. 2022, 59, 789-812.
10. Lee, S.H.; Kim, J.W.; Park, Y.S. The Influence of E-Learning on Exam Performance and the Role of Achievement Goals in Shaping Learning Patterns. Internet High. Educ. 2021, 50, 100-110.
11. Zhang, Y.; Li, X.; Wang, Q. Dynamic Personalized Learning Path Based on Triple Criteria Using Deep Learning and Rule-Based Method. IEEE Trans. Learn. Technol. 2020, 13, 283-294.
12. Chen, L.; Zhao, X.; Liu, H. Research on Dynamic Learning Path Recommendation Based on Social Network. Comput. Educ. 2019, 136, 1-10.
13. Nagisetty V.; Graves L.; Scott J.; Ganesh V. XAI-GAN: Enhancing Generative Adversarial Networks via Explainable AI Systems. arXiv preprint arXiv:2002.10438 2020.
14. Zhao, Y.; Sun, X.; Li, H.; Jin, Y. PerFed-GAN: Personalized Federated Learning via GAN. IEEE Trans. Neural Netw. Learn. Syst. 2021, 32, 3405-3415.
15. J. Balasubramanian, S. Koppaka, C. Rane, and N. Raul, ''Parameter based survey of recommendation systems,'' in Proc. Int. Conf. Innov. Comput. Commun. (ICICC), vol. 1, 2020. [Online]. Available: https://ssrn.com/abstract=3569579
16. A. S. M. Shibani, M. Mohd, A. T. A. Ghani, M. S. Zakaria, and S. M. Al-Ghuribi, ''Identification of critical parameters affecting an Elearning recommendation model using Delphi method based on expert validation,'' Information, vol. 14, no. 4, p. 207, Mar. 2023, doi: 10.3390/info14040207.
17. V. Vanitha, P. Krishnan, and R. Elakkiya, ''Collaborative optimization algorithm for learning path construction in Elearning,'' Comput. Electr. Eng., vol. 77, pp. 325–338, Jul. 2019, doi: 10.1016/j.compeleceng.2019.06.016.
18. A. H. Nabizadeh, D. Gonçalves, S. Gama, J. Jorge, and H. N. Rafsanjani, ''Adaptive learning path recommender approach using auxiliary learning objects,'' Comput. Educ., vol. 147, Apr. 2020, Art. no. 103777, doi: 10.1016/j.compedu.2019.103777.
19. S. Sarkar and M. Huber, ''Personalized learning path generation in Elearning systems using reinforcement learning and generative adversarial networks,'' in Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC), Oct. 2021, pp. 92–99, doi: 10.1109/SMC52423.2021.9658967.
20. Wang, D.; Zhang, T.; Liu, Y. FOKE: A Personalized and Explainable Education Framework Integrating Foundation Models, Knowledge Graphs, and Prompt Engineering. IEEE Access 2024, 12, 123456-123470.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 28 April 2025 doi:10.20944/preprints202504.2265.v1

21. Ogata, H.; Flanagan, B.; Takami, K.; Dai, Y.; Nakamoto, R.; Takii, K. EXAIT: Educational XAI Tools for Personalized Learning. Res. Pract. Technol. Enhanc. Learn. 2024, 19, 1-15.
22. Zhang, L.; Lin, J.; Borchers, C.; Cao, M.; Hu, X. 3DG: A Framework for Using Generative AI for Handling Sparse Learner Performance Data from Intelligent Tutoring Systems. arXiv preprint arXiv:2402.01746 2024.
23. Maity, S.; Deroy, A. Bringing GAI to Adaptive Learning in Education. arXiv preprint arXiv:2410.10650 2024.
24. Abbes, F.; Bennani, S.; Maalel, A. Generative AI in Education: Advancing Adaptive and Personalized Learning. SN Comput. Sci. 2024, 5, Article number: 1154.
25. Hariyanto, D.; Kohler, T. An Adaptive User Interface for an E-learning System by Accommodating Learning Style and Initial Knowledge. In Proceedings of the International Conference on Technology and Vocational Teachers (ICTVT 2017); Atlantis Press: 2017; pp. 16-2
26. Jeevamol, S.; Renumol, V.G.; Jayaprakash, S. An Ontology-Based Hybrid E-Learning Content Recommender System for Alleviating the Cold-Start Problem. Educ. Inf. Technol. 2021, 26, 7259–7283.
27. Elshani, L.; PirevaNuçi, K. Constructing a Personalized Learning Path Using Genetic Algorithms Approach. arXiv preprint arXiv:2104.11276 2021.
28. Sorva, J.; Sirkiä, A. Dynamic Programming - Structure, Difficulties and Teaching. Proc. 2013 IEEE Front. Educ. Conf. (FIE) 2013, 1685-1691.
29. Sharma, V.; Kumar, A. Smart Education with Artificial Intelligence Based Determination of Learning Styles. Procedia Comput. Sci. 2018, 132, 834-842.
30. Allioui, Y.; Chergui, M.; Bensebaa, T.; Belouadha, F.Z. Combining Supervised and Unsupervised Machine Learning Algorithms to Predict the Learners' Learning Styles. Procedia Comput. Sci. 2019, 148, 87-96.
31. Hmedna, B.; El Mezouary, A.; Baz, O.; Mammass, D. Identifying and Tracking Learning Styles in MOOCs: A Neural Networks Approach. Int. J. Innov. Appl. Stud. 2017, 19, 267–275.
32. Alghazzawi, D.; Said, N.; Alhaythami, R.; Alotaibi, M. A Survey of Artificial Intelligence Techniques Employed for Adaptive Educational Systems within E-Learning Platforms. J. Artif. Intell. Soft Comput. Res. 2017, 7, 47–64.
33. Nagisetty, V.; Graves, L.; Scott, J.; Ganesh, V. xAI-GAN: Enhancing Generative Adversarial Networks via Explainable AI Systems. arXiv preprint arXiv:2002.10438 2020.
34. Arslan, R.C.; Zapata-Rivera, D.; Lin, L. Opportunities and Challenges of Using Generative AI to Personalize Educational Assessments. *Front. Artif. Intell.* 2024, *7*, 1460651.
35. Schneider, J. Explainable Generative AI (GenXAI): A Survey, Conceptualization, and Research Agenda. *Artif. Intell. Rev.* 2024, *57*, 289.
36. Zhang, Y.-W.; Xiao, Q.; Song, Y.-L.; Chen, M.-M. Learning Path Optimization Based on Multi-Attribute Matching and Variable Length Continuous Representation. *Symmetry* 2022, *14*, 2360.
37. Yu, H.; Guo, Y. Generative Artificial Intelligence Empowers Educational Reform: Current Status, Issues, and Prospects. Front. Educ. 2023, 8, 1183162.
38. Carreon, A.; Goldman, S. Artificial Intelligence to Help Special Education Teacher Workload. Center for Innovation, Design, and Digital Learning 2024.
39. Sajja, R.; Sermet, Y.; Cikmaz, M.; Cwiertny, D.; Demir, I. Artificial Intelligence-Enabled Intelligent Assistant for Personalized and Adaptive Learning in Higher Education. Information 2024, 15, 596.
40. Jaboob, M.; Hazaimeh, M.; Al-Ansi, A.M. Integration of Generative AI Techniques and Applications in Student Behavior and Cognitive Achievement in Arab Higher Education. Int. J. Educ. Technol. High. Educ. 2024, 21, 45.
41. Mao, J.; Chen, B.; Liu, J.C. Generative Artificial Intelligence in Education and Its Implications for Assessment. TechTrends 2024, 68, 58–66.
42. Lai, J.W. Adapting Self-Regulated Learning in an Age of Generative Artificial Intelligence Chatbots. Future Internet 2024, 16, 218.
43. Pesovski, I.; Santos, R.M.; Henriques, R.; Trajkovik, V. Generative AI for Customizable Learning Experiences. Sustainability 2024, 16, 3034.
44. Liu, H.; Zhu, Y.; Wu, Z. Knowledge Graph-Based Behavior Denoising and Preference Learning for Sequential Recommendation. *IEEE Trans. Knowl. Data Eng.* 2024, *36*, 2490–2503.

doi:10.20944/preprints202504.2265.v1

45. Choi, H.; Lee, H.; Lee, M. Optimal Knowledge Component Extracting Model for Knowledge Concept Graph Completion in Education. *IEEE Access* 2023, *11*, 15002–15013.
46. Liang, X. Learning Personalized Modular Network Guided by Structured Knowledge. *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.* 2019, 8944–8952.
47. Sarkar, S.; Huber, M. Personalized Learning Path Generation in E-Learning Systems Using Reinforcement Learning and Generative Adversarial Networks. *Proc. IEEE Int. Conf. Syst. Man Cybern.* 2021, 1425–1432.