1. Citation

Saadia Gutta Essa, Turgay Celik, and Nadia Emelia Human-Hendricks (2023). "Personalized Adaptive Learning Technologies Based on Machine Learning Techniques to Identify Learning Styles: A Systematic Literature Review." *IEEE Access*, Volume 11, Pages 48392-48407.

2. Study Aim & Context

• **Main Goal:** To conduct a comprehensive **systematic literature review (SLR)** from 2015 to 2022 to identify emerging trends and gaps in the use of **Machine Learning (ML) techniques to automatically and dynamically identify learners' Learning Styles (LSs)** for personalized adaptive learning (PAL).

• **Focus:**

  ◦ **Intelligent tutoring systems (ITS)** and general e-learning platforms.

  ◦ **Feedback personalization** and personalized learning paths.

  ◦ **Automatic classification** of LS models (e.g., FSLSM, Kolb).

• **Field:** Computer Science, Educational Technology, and Artificial Intelligence.

3. Core ML Problem & Purpose

The research addresses the limitations of **statically determined Learning Styles**, which traditionally rely on time-consuming and often inaccurate questionnaires. The ML systems reviewed aim to:

• **Classify/Detect LSs automatically** based on student behavior and interactions.

• **Model learning behavior over time** to provide dynamic adaptivity.

• **Optimize the individual learning process** by mapping behavior to specific LS preferences.

4. Methodology & ML Approach

Aspect	Details
Research Type	**Systematic Literature Review (SLR)** of 48 primary studies.
Target Learners	Various, including primary students, high school, and university students across different e-learning platforms.

Data Used	**Log files (LF)** (73%), Static Information (SI), and other sources like image streaming or prior knowledge. Includes clickstreams, navigation data, and quiz scores.
ML Tasks	**Classification** (primary task), Clustering (e.g., K-means), and Regression.
Algorithms	Bayesian Networks (BN), **Artificial Neural Networks (ANN)**, Decision Trees (DT), Support Vector Machines (SVM), and Deep Learning (RNN, CNN).
Training Process	Includes train/test split and cross-validation; metrics used include **Accuracy, Precision, Recall, F-measure, and Kappa Statistics (KS)**.

5. Model Input Features

• **Knowledge Data (KD):** Correct/incorrect answers, quiz scores.

• **Chronometric Data (CD):** Time spent on tasks, reading material, or total session time.

• **Try Data (TD):** Number of attempts to determine correct answers.

• **Navigation Data (ND):** Frequency of topics selected, number of videos watched, and forum posts.

• **Feature Selection:** Features are often extracted from log files and mapped to specific dimensions of LS models like the **Felder-Silverman Learning Style Model (FSLSM)**.

6. Model Output & Interpretation

• **Model Prediction:** Predicted **Learning Style (LS)** category (e.g., Visual vs. Verbal, Active vs. Reflective).

• **Usage within System:**

◦ **Personalized learning contents and resources (LC):** Used in 17 of 48 studies to update web interface content.

◦ **Personalized learning paths (LP):** Used in 11 of 48 studies to adapt the sequence of materials.

◦ **Personalized interfaces (UI)** and **Intelligent Tutoring (TUT)**.

7. Algorithms / Techniques Compared

The sources identify a shift from single-algorithm models to hybrid and ensemble approaches to achieve higher accuracy in identifying Learning Styles (LS). While Decision Trees remain the most frequently used due to their interpretability, Artificial Neural Networks (ANN) and Deep

Learning (DL) are gaining significant traction for their ability to handle complex behavioral data.

A. Comparison of Machine Learning Methods

Algorithm	Pros Reported	Cons Reported / Frequency	Best Use Case
Decision Trees (DT)	**Highly interpretable; uses statistical metrics to branch alternatives and recognize new learners' LS.**	**Most frequently used (19 articles), but may overfit if attributes are not selected carefully.**	**Classifying learners based on discrete behavioral actions.**
Artificial Neural Networks (ANN)	**Capable of capturing complex patterns; gaining interest for enhancing efficiency and modeling dynamic LS.**	**Second most frequent (14 articles); often considered a "black box" compared to DT.**	**Modeling dynamic LS that change during the learning process.**
Bayesian Networks (BN)	**Effective at modeling uncertainty and probabilistic relationships in student interactions.**	**Third most frequent (13 articles); relies on probabilistic rather than deterministic classification.**	**Predicting LS when user data is incomplete or uncertain.**
Deep Learning (CNN/RNN)	**Provides intuitive algorithms for high-dimensional data; impacts MOOCs and e-learning significantly.**	**Limited empirical work exists documenting the comparison of different DL architectures.**	**Complex sequence modeling of learner behavior over time.**
Hybrid / Ensemble Methods	**Generally produces greater accuracy and more robust performance than single-algorithm models.**	**More computationally expensive; requires integrating multiple model outputs.**	**Achieving maximum precision in personalized learning paths.**

B. Evaluation Metrics Used

The efficacy and performance of these techniques were evaluated using a variety of statistical metrics to ensure the models accurately mapped behaviors to the Felder-Silverman Learning Style Model (FSLSM) or other frameworks.

• Accuracy (A), Precision (P), and Recall (R): The most common metrics for basic classification performance.

• F-measure (F1): Used to balance precision and recall in LS detection.

• Kappa Statistics (KS): Frequently cited to measure the agreement between predicted LS and actual learner behavior, often used to justify the superiority of one model over another.

• Root Mean Square Error (RMSE): Utilized to evaluate the error rate in predictive modeling.

C. Justification for Educational Data

The study justifies these models because Learning Styles are dynamic, not static. While traditional questionnaires provide a "snapshot," ML models (particularly ANNs and Hybrid systems) can track real-time changes in learner characteristics through log file attributes like Chronometric Data (time spent) and Navigation Data (frequency of topics selected). This allows the educational system to provide timely adaptivity that reduces cognitive overload.

8. Key Findings & Insights

• **ANNs are gaining significant attention** for their ability to enhance learning efficiency through LS identification.

• The **FSLSM** is the most frequently applied LS model because it characterizes learners in great detail across four dimensions.

• **Hybrid (combined) learning** approaches generally produce **greater accuracy** than single-algorithm models.

• **E-learning** remains the dominant platform (73%) for PAL research, followed by MOOCs (7%).

9. Pedagogical / Learning Implications

• **Personalization reduces cognitive overload** by providing optimal learning paths.

• **Self-regulation** is improved when students are aware of their LS, allowing them to capitalize on their strengths.

• The systems support **active learning** by matching content delivery to preferred student engagement methods (e.g., active vs. reflective).

10. Practical Design Implications for Auri's Journey

• **Prioritize Log File Analysis:** Use "Chronometric Data" (time) and "Try Data" (attempts) as primary features for real-time LS prediction.

• **Adopt the FSLSM Framework:** Use the Felder-Silverman dimensions as the target labels for the ML model, as they are well-suited for digital environments.

• **Implement Hybrid Models:** Combine different algorithms (e.g., DT and ANN) to achieve higher accuracy in classifying user behavior.

• **Focus on Learning Paths:** Use the ML output to not just change content, but to **dynamically adjust the learning path** (LP) for the user.

11. Strengths & Novel Contributions

• **Comprehensive Synthesis:** Aggregates findings from 48 influential studies to provide a roadmap of current ML trends in PAL.

• **Feature Taxonomy:** Clearly categorizes the types of educational data (KD, CD, TD, ND) that can be tracked for ML.

• **Platform Analysis:** Highlights which learning environments (e-learning, MOOC, ITS) are most stimulated by AI research.

12. Limitations & Identified Gaps

• **Deep Learning Gap:** There is a **lack of empirical investigation** comparing different deep learning algorithms for LS classification.

• **LS Model Diversity:** Most research is heavily skewed toward FSLSM, with limited exploration of other models like GTMI or MBTI in adaptive contexts.

• **Evaluation Metrics:** While accuracy is commonly measured, there is a need for more robust evaluations on **real-time adaptation** and long-term learning outcomes.

13. Hard ML Quotes / Definitions

• "Artificial intelligence (AI) approaches utilize algorithms from machine learning (ML) to tackle the challenge of personalising e-learning by mapping students' behavioural attributes to a particular LS automatically and dynamically".

• "LSs continually change during the learning process and are dynamic whereas results obtained from questionnaires are static".

14. Tags / Keywords

["learning_styles", "machine_learning", "FSLSM", "personalized_adaptive_learning", "ANN", "systematic_literature_review", "e-learning"]

15. Relevance Score & Subtype

• **ML Relevance:** 5/5

• **Educational Outcome Relevance:** 4/5

• **Applicability to Auri's Journey:** 5/5

• **Novelty for Thesis:** 4/5

16. Notes for Thesis Synthesis Section

Essa et al. (2023) demonstrate that **machine learning algorithms**, particularly **Artificial Neural Networks and Decision Trees**, can effectively replace static questionnaires for identifying student **learning styles (LSs)**. By analyzing **log file data**—such as time on task and navigation patterns—these systems can dynamically adjust **learning paths and content**, with the **Felder-Silverman model** serving as the most prevalent theoretical framework for such adaptations. Their review highlights that **hybrid ML models** offer superior accuracy, though a significant gap remains in the empirical comparison of **deep learning architectures** for real-time student modeling.

1. Citation

Sajja, R., Sermet, Y., Fodale, B., & Demir, I. (2023). "Evaluating AI-Powered Learning Assistants in Engineering Higher Education: Student Engagement, Ethical Challenges, and Policy Implications." *Working Paper/Preprint* (implemented at a large R1 public university).

2. Study Aim & Context

• **Main Goal:** To evaluate the **Educational AI Hub**, an AI-powered framework designed to support undergraduate engineering students through personalized learning and real-time support.

• **Focus:**

  ◦ **Intelligent Tutoring Systems:** Acting as an always-available virtual tutor.

  ◦ **Feedback Personalization:** Providing context-aware answers and semantic grading.

  ◦ **Student Engagement:** Examining perceptions of trust, ethics, and usability.

• **Field:** Engineering Education, Educational Technology, and Human-AI Interaction.

3. Core ML Problem & Purpose

The research addresses the need for **dynamic, 24/7 personalized support** in data-heavy disciplines like civil and environmental engineering. The ML system aims to:

• **Bridge communication gaps** outside of traditional classroom hours.

• **Synthesize complex academic sources** into concise, structured notes and flashcards.

• **Identify knowledge gaps** through generative quizzes and formative assessments.

• **Provide "semantic" grading**, moving beyond keyword matching to evaluate the underlying meaning of student responses.

4. Methodology & ML Approach

Aspect	Details
Research Type	**Mixed-methods** (Explanatory sequential design using surveys and usage logs).
Target Learners	Undergraduate civil and environmental engineering students (Sophomore and Senior levels).
Data Used	**System usage logs** (timestamps, session duration, interaction depth) and **Pre/Post surveys** (Likert-scale and open-ended).
ML Tasks	**Natural Language Processing (NLP)** for chatbot responses and **Generative AI** for content creation.
Algorithms	**Large Language Models (LLMs)**, specifically GPT-powered assistants and domain-specific embedding models (e.g., HydroLLM).
Training Process	Evaluation of a deployed system; focus on **semantic similarity** for grading and context-retention in chat.

5. Model Input Features

• **Interaction Type:** Chatbot messages vs. activity-based interactions (Notes, Quizzes, Flashcards).

• **Query Content:** Categorized into Software Help (40.4%), Conceptual/Theory Help (30.4%), and Assignment Help (19.5%).

• **Engagement Logs:** Frequency of repeat use and timestamps relative to assignment deadlines.

• **User Sentiments:** Survey data regarding trust, comfort, and perceived convenience.

6. Model Output & Interpretation

• **Model Prediction/Output:**

  ◦ **Context-aware responses** to student queries.

◦ **Bloom's Taxonomy-based flashcards** and quizzes.

◦ **Tiered coding assistance** (pseudocode first, then step-by-step explanations).

• **Usage within System:**

◦ **Adaptive Learning Paths:** Helping students navigate content at their own pace.

◦ **Formative Feedback:** Providing instant explanations and performance summaries after quizzes.

7. Algorithms / Techniques Compared

While this study focuses on the deployment of an AI framework rather than a head-to-head competition between classical ML classifiers, it evaluates the performance of **Large Language Models (LLMs)** and **domain-specific embedding models** against traditional **Human Assistance**.

A. Comparison of Methods: AI Assistant vs. Human Support

The researchers compared the **Educational AI Hub** (powered by LLMs like GPT-4 and HydroLLM) against traditional human instruction (TAs and Professors) across three primary dimensions: **Instructional Quality**, **Convenience**, and **Comfort**.

Entity / Method	Pros Reported	Cons Reported	Best Use Case
AI Hub (LLM / Generative AI)	**High Convenience (68%)**: Offers 24/7 availability. **Emotional Comfort (47%)**: Students feel less "judged" or intimidated.	**Low Instructional Quality (17%)**: Only a small minority felt it surpassed humans in teaching quality.	Immediate, low-stakes conceptual help and 24/7 technical support.
Human Support (TA/Prof)	**Superior Quality**: Perceived as having higher instructional authority and pedagogical depth.	**Inaccessibility**: Restricted by office hours. **Intimidation**: Can be perceived as unapproachable for "simple" questions.	High-stakes assessment and complex, nuanced instructional guidance.
Domain-Specific LLM (HydroLLM)	Optimized for **semantic retrieval** of academic documents (syllabi, textbooks).	Requires fine-tuning on specific course data to be effective.	Improving accuracy in discipline-focused educational answering.

B. Evaluation Metrics Used

The study utilized a mixed-methods evaluation to determine the effectiveness of the AI models:

• **5-Point Likert Scale:** Used to compare AI help to human help on a scale from “Much Worse” to “Much Better”.

• **Semantic Similarity:** A core ML metric used in the grading system to evaluate student responses against "ground truth" without requiring exact word matching.

• **Spearman Correlations (ρ):** Used to measure the relationship between student attitudes (trust, comfort) and actual system usage frequency.

• **Usage Logs:** Quantitative tracking of interaction depth, timestamps, and repeat use.

C. Model Suitability for Educational Data

The study justifies the use of **Generative AI (LLMs)** for this educational context because:

• **Adaptivity:** These systems can adapt to individual student needs, offering tailored feedback and real-time support that traditional static systems cannot.

• **Semantic Understanding:** Unlike keyword-matching tools, the AI Hub can interpret the underlying meaning of student inquiries and responses, supporting more authentic assessment (semantic grading).

• **Active Learning Support:** The model supports **Bloom’s Taxonomy** by generating tiered assistance, such as providing pseudocode before final solutions to encourage independent problem-solving.

8. Key Findings & Insights

• **Comfort is a major driver:** Students often feel less judged and more in control when interacting with an AI than a human instructor.

• **Task-specific relevance:** Students engaged most with AI when it supported **concrete outcomes** like homework completion (63%) or clarifying concepts (61%).

• **Trust is a barrier:** 58% of students worry about being accused of academic misconduct, which dampens full engagement.

• **Usage Patterns:** The chatbot was the primary interface (96% of users), while structured features like quizzes were used less frequently (31%).

9. Pedagogical / Learning Implications

• **Supplement, not replacement:** AI is most effective when framed as an academic aid rather than a shortcut.

• **Promotes Active Learning:** Features like flashcard generation and "semantic similarity" grading encourage students to express knowledge in their own words.

• **Self-Regulation:** AI assistants support independent problem-solving by providing tiered assistance (e.g., pseudocode before final answers).

10. Practical Design Implications for Auri's Journey

• **Focus on Usability & Accessibility:** Design the ML assistant to be "more convenient" than traditional resources to ensure high initial adoption.

• **Implement "Emotional Neutrality":** Leverage the fact that students feel more "comfortable" asking AI "stupid" questions to encourage deeper inquiry.

• **Clear Ethical Boundaries:** Provide explicit, course-level AI policies within the app to reduce student apprehension regarding academic integrity.

• **Task-Aligned Features:** Prioritize ML modules that assist with specific, high-friction tasks (like software troubleshooting).

11. Strengths & Novel Contributions

• **Real-world Deployment:** Evaluates AI in actual engineering courses rather than a laboratory setting.

• **Semantic Grading:** Introduces a nuanced approach to assessment that rewards understanding over exact wording.

• **Thematic Taxonomy:** Provides a clear breakdown of the types of help students seek from educational AI.

12. Limitations & Identified Gaps

• **Data Scale:** The sample was limited to 71 students in a single department, which may not generalize to all disciplines.

• **External AI Noise:** The study could not track if students were simultaneously using external tools like ChatGPT.

• **Institutional Uncertainty:** Highlights a "gray area" in policy that leaves students anxious about the ethical limits of AI use.

13. Hard ML Quotes / Definitions

• "Students regarded AI as a supplement rather than a replacement for human instruction".

• "The impact of AI depends heavily on how well it aligns with students' learning needs and how confidently students can engage with it in an ethically supported environment".

14. Tags / Keywords

["Generative_AI", "LLM", "Engineering_Education", "Student_Engagement", "Academic_Integrity", "Semantic_Grading", "Educational_AI_Hub"]

15. Relevance Score & Subtype

• **ML Relevance:** 4/5

• **Educational Outcome Relevance:** 5/5

• **Applicability to Auri's Journey:** 5/5

• **Novelty for Thesis:** 4/5

16. Notes for Thesis Synthesis Section

Sajja et al. (2023) highlight that **AI-powered assistants in higher education** serve as a critical supplement to human instruction by offering **high convenience and emotional comfort** for seeking help. While their LLM-based system successfully personalized learning through **semantic grading and real-time content synthesis**, the study underscores that **ethical uncertainty** regarding academic integrity remains a primary barrier to student engagement.

1. Citation

Yuto Yoshizawa and Yutaka Watanabe (2019). "Logic Error Detection System based on Structure Pattern and Error Degree." *Advances in Science, Technology and Engineering Systems Journal*, Vol. 4, No. 5, pp. 1-15.

2. Study Aim & Context

• **Main Goal:** To propose and evaluate a **logic error detection algorithm** based on structure patterns and "error degree" to support novice programmers.

• **Focus:**

◦ **Intelligent Tutoring Systems (ITS):** Acting as an autonomous environment for learning programming.

◦ **Feedback Personalization:** Providing automated, specific hints for debugging logic errors that do not cause program termination.

• **Field:** Computer Science, Programming Education, and e-Learning.

3. Core ML Problem & Purpose

The research addresses the **difficulty of debugging logic errors** (bugs where code runs but produces incorrect results) for novices. Traditional online judges only provide "Accepted" or "Wrong Answer" status without feedback on *why* the logic is flawed. The system aims to:

• **Identify logic errors automatically** by comparing student code to correct examples.

• **Classify error types** (e.g., loop conditions, calculation errors).

• **Model the "Error Degree"** to select the most appropriate correct code for comparison to provide the best feedback.

4. Methodology & ML Approach

Aspect	Details
Research Type	**Quantitative and Qualitative** experimental evaluation using a deployed API.
Target Learners	**Novice programmers** (university students) using the Aizu Online Judge (AOJ).
Data Used	**Source code (Java)** from 60,000 users and 4 million submissions stored in the AOJ database.
ML Tasks	**Classification** of error types and **Pattern Matching** based on code structure.
Algorithms	**Abstract Syntax Trees (AST)** for recursive structural comparison and a **weighted similarity algorithm** for "Error Degree."
Training Process	Evaluation using **Whole Success Ratio** and **Partial Success Ratio** across a dataset of varied programming problems (Intro to Programming set).

5. Model Input Features

• **Structure Patterns:** Syntax sequences (e.g., Variable Declaration → For Statement → Method Call) generated from AST depth-first searches.

• **Terminal Constructs:** Specific information within code blocks, such as variable names or literal values.

• **Nonterminal Constructs:** Grammatical structures like *if* statements or *for* loops.

• **Operator Data:** Operators used in expressions (mapped to specific weights for error degree).

6. Model Output & Interpretation

• **Model Prediction:**

◦ **Error Classification:** Identified logic errors such as "Loop condition," "Conditional branch," or "Calculation" errors.

◦ **Error Degree:** A numerical value representing the index of similarity to a correct solution.

• **Usage within System:**

◦ **Provide Personalized Feedback:** Generating messages like "Wrong index accessed" or "Incorrect rounding" based on the comparison with the lowest error degree.

◦ **Autonomous Learning Support:** Allowing students to debug without a human teacher.

7. Algorithms / Techniques Compared

The study justifies the **AST-based approach** by comparing it to other source code analysis methods.

Algorithm / Method	Pros Reported	Cons Reported	Best Use Case
Text-based Comparison	Simple to implement.	Includes unique info like variable names; poor at finding logic.	Plagiarism detection.
Token-based Comparison	High accuracy for lexical analysis.	Does not necessarily reflect the **structure** of the code.	Basic syntax checking.
AST-based (Proposed)	Captures **grammatical structure**; suitable for logic error detection.	Comparison of different tree shapes (different algorithms) can be difficult.	**Logic error feedback.**
Semantic-based (Graphs)	Uses data/control dependency info.	Hard to detect logic errors that differ only in internal values.	Deep logic analysis.

8. Key Findings & Insights

• **High Detection Accuracy:** The algorithm achieved a **Whole Success Ratio** (detecting all logic errors) of over **70%** for many problems and a **Partial Success Ratio** of over **80%**.

• **Effective Feedback:** Qualitative tests showed that **80.43%** of the generated feedback was "appropriate" for supporting student learning and debugging.

• **Structure Importance:** Filtering by **Structure Pattern** significantly improves the relevance of the correct code used as a comparison target.

9. Pedagogical / Learning Implications

• **Promotes Learner Autonomy:** By providing feedback without a teacher, the system supports self-directed "autonomous learning."

• **Reduces Frustration:** Specifically targets logic errors that cause novices to "give up" because they cannot find bugs in code that otherwise compiles.

• **Active Learning:** Encourages students to fix their own errors based on hints rather than just giving them the correct answer.

10. Practical Design Implications for Auri's Journey

• **AST Analysis:** Utilize **Abstract Syntax Trees** rather than raw text to understand user logic or block-based code patterns.

• **Weighted Feedback (Error Degree):** Implement a **weighting system** (e.g., high weight for syntax-level errors, lower for terminal construct errors) to prioritize feedback.

• **Structural Filtering:** When comparing a user's attempt to a "master" solution, first **filter by structural similarity** to ensure the feedback is relevant to the user's specific approach.

11. Strengths & Novel Contributions

• **Error Degree Metric:** A novel way to quantify and prioritize code similarity for feedback.

• **Scalable API:** The development of the **LED (Logic Error Detector) API** allows this logic to be integrated into various e-learning platforms.

• **Real-world Data:** Tested against a massive dataset from the **Aizu Online Judge**, proving its effectiveness in a large-scale educational environment.

12. Limitations & Identified Gaps

• **Algorithmic Flexibility:** The system struggles to compare codes that use **entirely different algorithms** (e.g., a *for* loop vs. a *while* loop) for the same problem.

• **Language Specificity:** Currently focused and evaluated on **Java**; generalization to other languages (though possible) was not the primary focus.

• **"Ground Truth" Dependency:** The system requires a large database of "Accepted" codes to find a suitable comparison target.

13. Hard ML Quotes / Definitions

• "A logic error is a bug in a program that triggers erroneous behavior but does not cause abnormal termination."

• "Structure pattern can be considered to express the general form of source code."

14. Tags / Keywords

["logic_error_detection", "AST", "programming_education", "automated_feedback", "Aizu_Online_Judge", "error_degree", "structure_patterns"]

15. Relevance Score & Subtype

• **ML Relevance:** 5/5

• **Educational Outcome Relevance:** 5/5

• **Applicability to Auri's Journey:** 5/5

• **Novelty for Thesis:** 4/5

16. Notes for Thesis Synthesis Section

Yoshizawa and Watanabe (2019) demonstrate that **logic error detection** can be significantly improved by utilizing **Abstract Syntax Trees (AST)** to identify "Structure Patterns" and calculating an "**Error Degree**" for weighted comparison. Their system successfully provided appropriate debugging feedback in **over 80% of cases** by matching student code to the most structurally similar correct solutions, suggesting that **structural similarity** is a more effective feature than text-based metrics for personalizing feedback in technical learning environments.

1. Citation

Heeryung Choi, Tung Phung, Mengyan Wu, Adish Singla, and Christopher Brooks (2025). "Reflection-Satisfaction Tradeoff: Investigating Impact of Reflection on Student Engagement with AI-Generated Programming Hints." *arXiv preprint arXiv:2512.04630v1 [cs.CY]*.

2. Study Aim & Context

• **Main Goal:** To investigate how different designs and placements of **reflection prompts** affect student engagement, satisfaction, and learning quality when using **AI-generated programming hints**.

• **Focus:**

◦ **Intelligent Tutoring Systems:** Providing timely, personalized debugging support.

◦ **Feedback Personalization:** Pairing AI-generated hints with reflection prompts.

◦ **Self-Regulated Learning (SRL):** Scaffolding metacognitive skills during problem-solving.

• **Field:** Computer Science (Programming Education) and Educational Artificial Intelligence.

3. Core ML Problem & Purpose

The research addresses **"metacognitive laziness,"** a tendency for students to disengage from effortful learning when on-demand AI assistance is available. The system aims to:

• **Leverage Generative AI (GPT-4)** to provide personalized debugging hints without compromising deep learning.

• **Identify the "Reflection-Satisfaction Tradeoff,"** where optimizing for user satisfaction may undermine the pedagogical goal of effortful learning.

• **Model reflective behavior** across different SRL phases (planning, monitoring, evaluation).

4. Methodology & ML Approach

Aspect	Details
Research Type	**Quantitative and Qualitative** field experiments (Two-trial randomized controlled trials).
Target Learners	**Adult learners** in an online Master's degree program in applied data science.
Data Used	**Log files** (hint requests, submissions), timestamps, student text reflections, and hint satisfaction ratings.
ML Tasks	**Generative AI** for hint production and **Thematic Analysis** for reflection quality coding.
Algorithms	**GPT-4** (engine for hints), 5Rs framework for reflection levels, and thematic qualitative coding.
Training Process	Evaluation via **Immediate Success Rate**, satisfaction rates, and expert-coded reflection quality.

5. Model Input Features

• **Buggy Code & Output:** The student's current code and the interpreter's error messages were fed into GPT-4.

• **SRL Phase Prompts:** Inputs categorized by phase: **Planning** (approaching the problem), **Monitoring** (tracking progress), and **Evaluation** (assessing the hint).

• **Prompt Guidance Level: Directed** (specific guidance) vs. **Open** (minimal guidance).

• **Placement Timing:** Prompts delivered either **before** or **after** receiving the AI hint.

6. Model Output & Interpretation

• **Model Prediction/Output:**

◦ **Personalized Hints:** AI-generated debugging guidance tailored to the student's specific bug.

◦ **Reflection Quality:** Thematic categorization of student responses (What, Why, How).

• **Usage within System:**

◦ **Triggering Scaffolds:** Prompts were used to force a pause in the help-seeking process to encourage critical thinking.

◦ **Adaptivity:** The system collected student reflections to potentially improve future hint relevance and quality.

7. Algorithms / Techniques Compared

The study compared different **interaction designs** rather than just ML classifiers.

Condition / Technique	Pros Reported	Cons Reported	Best Use Case
Before-hint Reflection	**Higher-quality reflections**; encourages deeper planning.	**Lower satisfaction**; perceived as higher friction.	Fostering deep learning/SRL.
After-hint Reflection	Easier for students; allows rating of the hint itself.	Reflections were often superficial or focused on hint accuracy.	Feedback on AI performance.
Directed Prompts	Produced **more constructive reflections** that included "how" components.	Reduced student satisfaction.	Novice learners requiring structure.
Open Prompts	Higher satisfaction and lower friction.	Less constructive; reflections often skipped or brief.	High-autonomy learners.

8. Key Findings & Insights

• **Inverse Relationship:** Students who produced the highest-quality reflections reported the **lowest satisfaction** with the AI hints.

• **SRL Impact: Planning-focused** prompts led to the most sophisticated reflections compared to monitoring or evaluation.

• **Performance Neutrality:** Immediate performance (success rates) did not significantly differ across conditions, suggesting AI can support immediate task completion while undermining long-term skill development if not carefully designed.

• **Metacognitive Laziness:** Without prompts, students often "delegate" thinking to the AI, reducing deep learning opportunities.

9. Pedagogical / Learning Implications

• **Productive Struggle:** Aligns with the concept of **"desirable difficulties,"** suggesting that learning is most effective when it requires effort, even if it lowers user satisfaction.

• **Constructivist Alignment:** By requiring reflection *before* help, the system forces students to actively construct their own understanding of their errors.

• **Systemic Tradeoff:** Current AI development prioritizes user convenience, which directly conflicts with the pedagogical need for **effortful learning**.

10. Practical Design Implications for Auri's Journey

• **Design for Friction:** Implement reflection prompts **before** providing AI solutions to ensure the user processes the error themselves.

• **Directed Scaffolding:** Use **directed prompts** (asking "why" or "how") to help users analyze their problem-solving steps rather than just describing the error.

• **Multi-Metric Evaluation:** Do not rely solely on "satisfaction" or "NPS" scores; track **reflection quality** and **metacognitive engagement** to measure true learning impact.

11. Strengths & Novel Contributions

• **Field Evidence:** Provides empirical data from an **authentic learning environment** (a Master's level course) rather than a lab setting.

• **Thematic Depth:** Uses a nuanced coding framework (5Rs) to measure the *quality* of human-AI interaction.

• **Novel Framework:** Introduces the **"Reflection-Satisfaction Tradeoff"** as a critical concept for educational AI design.

12. Limitations & Identified Gaps

• **Sample Size:** Trial 1 had a relatively small cohort (74 students), limiting some statistical power.

• **Long-term Outcomes:** The study focused on immediate success rates; **long-term retention** and skill transfer were not measured.

• **Human-in-the-Loop:** Did not incorporate tutors or instructors to see how AI reflection compares to human-led reflection.

13. Hard ML Quotes / Definitions

• "**Metacognitive laziness**: the tendency of students to avoid and disengage from effortful learning processes [when using AI]".

• "Prioritizing satisfaction alone in AI optimization risks diminishing opportunities for students to **'struggle' productively** and develop SRL skills".

14. Tags / Keywords

["AI-generated_hints", "metacognitive_laziness", "reflection-satisfaction_tradeoff", "SRL", "generative_AI", "educational_AI_design", "desirable_difficulties"]

15. Relevance Score & Subtype

• **ML Relevance:** 4/5 (Focuses on LLM interaction design)

• **Educational Outcome Relevance:** 5/5

• **Applicability to Auri's Journey:** 5/5

• **Novelty for Thesis:** 5/5

16. Notes for Thesis Synthesis Section

Choi et al. (2025) reveal a fundamental **"Reflection-Satisfaction Tradeoff"** in AI-assisted learning, where the designs that foster the highest quality of metacognitive reflection—such as **directed prompts placed before a hint**—result in the lowest student satisfaction. Their findings caution that current AI optimization for user convenience may exacerbate **"metacognitive laziness,"** and they argue for a shift in educational AI design toward prioritizing **"productive struggle"** over immediate user satisfaction to ensure long-term skill development.

1. Citation

Sein Minn (2021). "BKT-LSTM: Efficient Student Modeling for knowledge tracing and student performance prediction." *arXiv preprint arXiv:2012.12218v3*.

2. Study Aim & Context

• **Main Goal:** To propose an efficient student model called **BKT-LSTM** that combines the psychologically meaningful parameters of **Bayesian Knowledge Tracing (BKT)** with the predictive power of **Long Short-Term Memory (LSTM)** neural networks.

• **Focus:**

◦ **Knowledge Tracing (KT):** Tracing the knowledge state of students dynamically.

◦ **Skill Prediction:** Predicting if a student will answer a problem correctly based on past interactions.

◦ **Adaptive Learning Environments:** Providing a foundation for individualization and personalization in Intelligent Tutoring Systems (ITS).

• **Field:** Computer Science, Artificial Intelligence, and Learning Analytics.

3. Core ML Problem & Purpose

The research addresses the limitations of two dominant KT methods: BKT, which ignores **learning transfer** across skills, and Deep Knowledge Tracing (DKT), which lacks **psychologically meaningful interpretation**.

• **Primary Task:** Predicting student performance in sequential interaction data.

• **Specific Goal:** Modeling the student's state of conceptual or procedural knowledge from observed performance on tasks while accounting for **problem difficulty** and individual **learning ability**.

4. Methodology & ML Approach

Aspect	Details
Research Type	**Quantitative / Experimental** using large-scale public datasets.
Target Learners	**Middle and High school students** (via ASSISTments) and **University students** (via Cognitive Tutor Algebra dataset).
Data Used	**Log files** containing student IDs, skill IDs, problem IDs, and binary response outcomes (correct/incorrect).
ML Tasks	**Classification** (predicting binary outcomes) and **Clustering** (detecting ability profiles).
Algorithms	**BKT-LSTM** (Proposed hybrid), BKT, DKT, DKVMN (Dynamic Key-Value Memory Networks), PFA (Performance Factors Analysis), and BIRT (Bayesian IRT).
Training Process	**5-fold cross-validation**; evaluated using **AUC** (Area Under Curve), **RMSE** (Root Mean Squared Error), and $r2$.

5. Model Input Features

• **Skill Mastery:** Assessed by a standard BKT Markov model to infer the probability that a student knows a specific skill.

• **Ability Profile:** Detected via **k-means clustering** of a student's success rates across all skills to capture "learning transfer" (how well a student applies knowledge to new skills).

• **Problem Difficulty:** Calculated based on the average success rate of all students who attempted a specific problem, mapped onto a scale of 1 to 10.

• **Interaction History:** Sequential data of past student attempts preserved through LSTM hidden layers.

6. Model Output & Interpretation

• **Model Prediction:** The probability (*yt*) that a student will answer a particular problem associated with a specific skill correctly at the next time step (*t*+1).

• **Usage within System:**

◦ **Personalization:** Optimizing instruction by identifying which concepts a student has mastered vs. where they struggle.

◦ **Adaptive Feedback:** Enabling tutoring systems to provide support based on the predicted likelihood of student failure.

7. Algorithms / Techniques Compared

The study justified BKT-LSTM by comparing it against several state-of-the-art models across three datasets.

Algorithm	Pros Reported	Cons Reported	Best Use Case
BKT	Meaningful parameters (guess/slip).	Cannot model learning transfer across different skills.	Simple skill modeling.
DKT (LSTM)	Preserves past info in sequence; better prediction than BKT.	"Black box"; thousands of parameters with no psychological meaning.	Complex sequence prediction.
BIRT / IRT	Strong theoretical background in psychometrics.	Static; cannot perform cognitive diagnosis easily.	Standardized testing.
BKT-LSTM	**Outperforms all models** (AUC ~0.85 on Algebra); provides interpretable features.	Requires clustering step for ability profiles.	**High-precision KT.**

8. Key Findings & Insights

• **Superior Performance:** BKT-LSTM showed a notable gain of **10% in AUC** compared to standard DKT and DKVMN in the ASSISTments09 dataset.

• **Value of Difficulty:** Ablation studies showed that adding **problem difficulty** to the neural network increases prediction accuracy by **8 to 10%**.

• **Learning Transfer:** The "ability profile" (cluster ID) captures student learning evolution over time, confirming that learning transfer is a critical factor in performance prediction.

9. Pedagogical / Learning Implications

• **Individualization:** By tracing knowledge states dynamically, the system allows for the optimization of instruction to meet individual needs.

• **Refinement of Mastery:** Moving beyond binary "mastered/not mastered" to a probabilistic value of skill mastery allows for more nuanced tutoring.

• **Support for Active Learning:** The model facilitates "remediate current performance" by automatically identifying where students need more practice.

10. Practical Design Implications for Auri's Journey

• **Hybridize Models:** Combine traditional educational models (like BKT) with modern deep learning (LSTM) to get both **interpretability and accuracy**.

• **Cluster for Ability:** Use **k-means clustering** to group users by their learning pace or "ability profile" to better predict how they will handle new topics.

• **Scale Problem Difficulty:** Don't treat all problems within a skill as equal; calculate and feed **item-level difficulty** into the ML model.

11. Strengths & Novel Contributions

• **Efficient Hybridization:** Successfully integrates latent variables from psychometrics (mastery/difficulty) into a deep learning architecture.

• **Temporal Ability Tracking:** Detects "ability profiles" across time intervals rather than assuming a static student ability.

• **Robust Evaluation:** Validation across three distinct, widely recognized public datasets.

12. Limitations & Identified Gaps

• **Data Density:** The model requires a minimum number of attempts (e.g., 20) before it can start accurately assigning an "ability profile".

• **Computational Cost:** Training LSTMs with large numbers of skills and students requires more resources than simpler models like PFA or BKT.

• **Parameter Independence:** In original BKT, skills are learned independently; while BKT-LSTM addresses this, it still relies on pre-defined skill mappings (Q-matrices).

13. Hard ML Quotes / Definitions

• "**Knowledge Tracing (KT)** is the assessment of students' knowledge state dynamically and the prediction of whether students may or may not answer a problem correctly based on past item test outcomes".

• "**Skill mastery** is not binary value and which is the probability of learning skill *st* rather than the probability of student applying the skill correctly".

14. Tags / Keywords

["knowledge_tracing", "BKT-LSTM", "student_modeling", "learning_transfer", "deep_learning", "problem_difficulty", "AUC_performance"]

15. Relevance Score & Subtype

• **ML Relevance:** 5/5

• **Educational Outcome Relevance:** 5/5

• **Applicability to Auri's Journey:** 5/5

• **Novelty for Thesis:** 5/5

16. Notes for Thesis Synthesis Section

Minn (2021) demonstrates that the **BKT-LSTM hybrid model** significantly improves the accuracy of student performance prediction by integrating **psychologically meaningful features**—such as skill mastery and problem difficulty—into a deep learning framework. By utilizing **k-means clustering** to identify dynamic **ability profiles**, the model successfully accounts for **learning transfer across skills**, achieving a 10% improvement in AUC over traditional deep knowledge tracing and providing a more interpretable roadmap for personalized adaptive instruction.

1. Citation

Tung Phung, Mengyan Wu, Heeryung Choi, Gustavo Soares, Sumit Gulwani, Adish Singla, and Christopher Brooks (2025). "Bridging Gaps Between Student and Expert Evaluations of AI-Generated Programming Hints." *Proceedings of the Twelfth ACM Conference on Learning @ Scale (L@S '25)*, July 21–23, 2025, Palermo, Italy.

2. Study Aim & Context

• **Main Goal:** To systematically study the mismatches in perceived quality of AI-generated hints from the perspectives of both students and expert educators.

• **Focus:**

◦ **Intelligent Tutoring Systems:** Using GPT-4 to provide personalized feedback at scale.

◦ **Feedback Personalization:** Investigating how to align AI hints with students' perceived helpfulness.

◦ **Evaluation Rubrics:** Investigating the alignment between expert-designed scoring rubrics and student ratings.

• **Field:** Computing education, Generative AI, and Human-AI interaction.

3. Core ML Problem & Purpose

The research addresses the challenge of ensuring that automated feedback is both **pedagogically sound** (from an expert view) and **perceived as helpful** (by the student). The ML system aims to:

• **Generate Socratic-style programming hints** using Generative AI (GPT-4).

• **Bridge evaluation discrepancies** where experts rate a hint as high-quality but students find it unhelpful.

• **Personalize feedback** by incorporating symbolic information and student-provided "thoughts" into the prompt.

4. Methodology & ML Approach

Aspect	Details
Research Type	**Mixed-methods**: Quantitative comparison of student/expert ratings and Qualitative categorization of mismatch reasons.

Target Learners	**Adult learners** in an online Master's program (introductory data science course).
Data Used	**Log files** from a JupyterLab extension, student ratings (Helpful/Unhelpful), and optional student "thoughts" on their issues.
ML Tasks	**Generative AI** for feedback generation and **Classification** of hint quality based on rubrics.
Algorithms	**GPT-4** (used for hint generation, Chain-of-Thought explanations, and generating fixed programs).
Training Process	Evaluation via **Chi-square (*X*2) tests** to compare rating agreement and 2x2 contingency tables.

5. Model Input Features

• **Symbolic Information:** The student's **buggy program**, the **problem description**, and the **buggy output** generated by running the student's code.

• **Reference Solutions:** A **fixed program** generated by the AI to help identify the necessary corrections.

• **Proposed Interactive Features:**

◦ **Pre-hint Thoughts:** The student's own assessment of their issue.

◦ **Trajectory/History:** Information from previous hint requests for the same question.

◦ **Overarching Bug:** Instructing the model to prioritize the most critical conceptual bug.

6. Model Output & Interpretation

• **Model Prediction/Output:** A **Socratic-style hint** limited to a sentence or two, designed to guide the student without providing the final answer or code.

• **Usage within System:**

◦ **Triggering Feedback:** Hints are provided upon a student clicking a "Hint" button in their Jupyter notebook.

◦ **Adaptive Refinement:** The system uses the "pre-hint thoughts" to refine the prompt, aiming to address the student's specific concern rather than just a low-level code error.

7. Algorithms / Techniques Compared

The study compares the **Deployed Technique** (standard GPT-4 prompt) against an **Adjusted Technique** incorporating student interaction data.

Method	Pros Reported	Cons Reported	Best Use Case
Deployed Technique (GPT-4)	Uses Chain-of-Thought; performs well in general data science contexts.	Can be **overly optimistic**; misses the student's specific conceptual confusion.	General debugging assistance.
Adjusted Technique (+Thoughts/History)	Addresses mismatch cases; aligns better with student needs by acknowledging their specific progress and trajectory.	Requires more complex prompt engineering and user input.	**Personalized, high-precision tutoring.**

8. Key Findings & Insights

• **Significant Rating Mismatch:** Disagreement between students and experts occurred in **34.5% of cases**.

• **Expert Optimism:** 77% of mismatches were cases where experts rated a hint as "high-quality" while students found it "unhelpful".

• **Five Mismatch Categories:** Reasons for discrepancies include ignoring the student's **trajectory**, ignoring their **solving approach**, or failing to address the student's **specific concern**.

• **Rubric Alignment:** Augmenting the expert rubric with attributes like "Accounting for student's concern" and "Informative given history" led to a higher alignment with student perceptions.

9. Pedagogical / Learning Implications

• **Metacognitive Alignment:** Effective AI feedback must recognize the **cognitive demands** of the task as perceived by the student.

• **Acknowledgment of Progress:** Providing hints that only state what is "incomplete" can be unhelpful if the student is already on the right track; the system must acknowledge their progress.

• **Socratic vs. Direct Aid:** There is a mismatch in pedagogical objectives; while experts prefer Socratic hints to foster reasoning, students often desire more detailed assistance.

10. Practical Design Implications for Auri's Journey

• **Incorporate User Context:** Use a "Pre-hint Thoughts" prompt to allow the user to describe their struggle, focusing the ML model on the user's **actual confusion** rather than just the code's syntax.

• **Track Interaction History:** Ensure the ML prompt includes **previous requests (Trajectory)** so the AI doesn't repeat information the user has already rejected or understood.

• **Prioritize Critical Bugs:** Use the "Overarching Bug" instruction to ensure the AI addresses **conceptual misunderstandings** before minor typos.

11. Strengths & Novel Contributions

• **Holistic Evaluation:** Combines student and expert perspectives to avoid the "biased evaluations" found in studies that only use one group.

• **Extended Rubric:** Proposes new quality attributes (e.g., "Tackling overarching bug") to make expert evaluations more pedagogically sound and student-aligned.

• **Interactive Prompting:** Demonstrates that incorporating **human thoughts** into the AI prompt can effectively bridge the gap between human and machine logic.

12. Limitations & Identified Gaps

• **Small Sample Size:** Only 34 students requested hints, limiting generalizability.

• **Single Context:** Conducted in only one Python programming course.

• **Validation Gap:** The effectiveness of the proposed adjustments on **long-term learning outcomes** still needs to be validated through larger classroom deployments.

13. Hard ML Quotes / Definitions

• "Ensuring that automated feedback benefits learning is non-trivial as effective feedback should be both pedagogically sound from expert educators' perspectives and perceived as helpful by students".

• "Rubric-based expert evaluations may be **overly optimistic**, necessitating further diagnosis and improvement".

14. Tags / Keywords

["AI-generated_hints", "GPT-4", "student_modeling", "feedback_personalization", "rubric_evaluation", "Socratic_hints", "mismatch_analysis"]

15. Relevance Score & Subtype

• **ML Relevance:** 4/5

• **Educational Outcome Relevance:** 5/5

• **Applicability to Auri's Journey:** 5/5

• **Novelty for Thesis:** 5/5

16. Notes for Thesis Synthesis Section

Phung et al. (2025) highlight a critical **34.5% discrepancy** between student and expert evaluations of AI hints, noting that experts often overrate hints that students find unhelpful. By identifying mismatch categories—such as the AI ignoring a student's previous requests or their specific conceptual concerns—they suggest that **integrating student pre-hint thoughts** and **interaction trajectory** into LLM prompts can create a more personalized and effective "Socratic" feedback loop.

1. Citation

Rastrollo-Guerrero, J. L., Gómez-Pulido, J. A., & Durán-Domínguez, A. (2020). "Analyzing and Predicting Students' Performance by Means of Machine Learning: A Review." *Applied Sciences*, Volume 10, Issue 3, 1042.

2. Study Aim & Context

• **Main Goal:** To provide an in-depth overview of modern techniques and algorithms (Machine Learning, Collaborative Filtering, Recommender Systems, and Artificial Neural Networks) applied to predict student performance and behavior.

• **Focus:**

  ◦ **Skill Prediction & Performance:** Predicting grades and academic results.

  ◦ **Dropout Prevention:** Identifying at-risk students to plan interventions.

◦ **Intelligent Systems:** Automation of student activity analysis via technology-enhanced learning tools.

◦ **Recommendation:** Suggesting activities and resources to improve student experience.

• **Field:** Artificial Intelligence, Machine Learning, and Data Mining in Education.

3. Core ML Problem & Purpose

The research analyzes how ML addresses the need to predict future student behavior to improve curriculum design and guidance. Specific problems addressed include:

• **Predicting student mastery/performance:** Modeling grades and knowledge levels.

• **Detecting dropout risk:** Identifying students likely to leave, particularly in early stages (first two years).

• **Classifying performance:** Grouping students into High, Medium, or Low performance categories.

• **Knowledge discovery:** Analyzing patterns in students' learning styles.

4. Methodology & ML Approach

Aspect	Details
Research Type	**Qualitative Review** of 64 recent peer-reviewed articles (mostly published between 2014–2020).
Target Learners	University (70%), High School (16%), and School (12%).
Data Used	**Log files**, demographic characteristics, grades, interaction events (e.g., Moodle), and massive volumes of technology-enhanced learning data.
ML Tasks	**Supervised Learning** (50%), Collaborative Filtering (26%), Artificial Neural Networks (11%), and Unsupervised Learning (5%).
Algorithms	SVM, Decision Trees, Naïve Bayes, Random Forest, ANN, KNN, Deep Learning, Logistic Regression, and Matrix Factorization.
Training Process	Comparison of various models across studies using **train/test split** and **cross-validation** evaluated by accuracy, RMSE, and recovery rate.

5. Model Input Features

• **Academic Data:** Grades from tasks, exams, cumulative GPA, and previous course results.

• **Demographics:** Gender, ethnicity, and geographic environment.

• **Social & Personal:** Family history, socioeconomic status, parental background, and student "thoughts" or learning situation descriptions.

• **Engagement Logs:** Interaction events with Moodle modules, time spent on tasks, and consistency in using online resources.

• **Psychological/Cognitive:** Cognitive and non-cognitive measures, learning styles, and behavior patterns.

6. Model Output & Interpretation

• **Model Prediction:**

◦ **Performance Levels:** High, Medium, or Low performance classification.

◦ **Outcome Probabilities:** Probability of student dropout or successfully obtaining a certificate.

◦ **Continuous Metrics:** Final grades and GPA at graduation.

• **Usage within System:**

◦ **Trigger Intervention:** Early identification of at-risk students for academic support.

◦ **Adaptive Content:** Suggesting resources and activities to help new students based on old students' experiences.

◦ **Decision Support:** Justifying educational approaches and informing decision-making for instructors.

7. Algorithms / Techniques Compared

The review tracks which algorithms were applied and which performed best across the 64 studies.

Algorithm	Pros Reported	Cons Reported	Best Use Case
SVM	**Highest accuracy** in many cases (up to 97.98%); best performance for GPA and retention.	Requires balanced data (often used with SMOTE).	Retention and grade prediction.
Naïve Bayes	Most appropriate for distance learning performance/dropout in some contexts.	May be outperformed by more complex models.	Early diagnostic prediction.

Decision Tree	Interpretable; accuracy improves when cognitive traits are included.	Can be less accurate than SVM or ANN in complex scenarios.	Behavior/Cognitive analysis.
ANN	Great precision in predicting performance; models non-linear relationships.	Less interpretable; less used than supervised ML.	High-precision performance prediction.
Unsupervised (Clustering)	Can group students into performance-based paths.	**Unattractive due to low accuracy** in predicting behavior.	Performance-based grouping.

8. Key Findings & Insights

• **Dominance of Supervised ML:** Supervised learning is the most widely used (50%) and reliable technique for behavioral prediction.

• **Top Performers: SVM** was the most used and provided the most accurate results overall, followed by Decision Trees, Naïve Bayes, and Random Forest.

• **Critical Timing:** More than 60% of dropouts occur in the first two years, making early identification (e.g., by the third week) vital.

• **Accuracy Boosters:** Pre-processing and data balancing (like SMOTE) significantly improve prediction accuracy for unbalanced datasets.

• **Unsupervised Gap:** Unsupervised learning is currently under-researched in education due to lower predictive reliability compared to other methods.

9. Pedagogical / Learning Implications

• **Informed Decision-Making:** ML models provide relevant data to facilitate teacher decision-making and curriculum planning.

• **Resource Efficiency:** Early identification of failing students saves government resources and educator effort.

• **Personalization:** Adaptive feedback and resource recommendation improve the individual student experience.

• **Proactive Retention:** Identifying the "why" behind dropout (e.g., socioeconomic status) allows for preventative social or academic mechanisms.

10. Practical Design Implications for Auri's Journey

• **Implement SVM for Performance:** Prioritize **Support Vector Machines** as a primary classifier for high-accuracy performance prediction.

• **Early-Stage Modeling:** Focus ML tracking on the **initial weeks** of a course to provide early warning signals.

• **Feature Diversity:** Include **non-cognitive and cognitive features** (like behavior patterns and learning styles) to increase model accuracy.

• **Balance Data:** Always apply data balancing techniques (like **SMOTE**) when handling unbalanced student interaction datasets to avoid misleading accuracy.

11. Strengths & Novel Contributions

• **Comprehensive Synthesis:** Aggregates findings from 64 distinct papers to identify industry-wide trends in EDM.

• **Multi-Level Analysis:** Evaluates ML effectiveness across different education levels (School to University).

• **Dual Classification:** Organizes research by both the ML **technique** and the pedagogical **objective**.

12. Limitations & Identified Gaps

• **Education Level Bias:** 70% of research focuses on University; there is a **gap in ML application at the primary school level**.

• **Small-Scale Challenges:** Collaborative filtering accuracy drops significantly in **small university settings** or courses with few students.

• **Unsupervised Learning:** Opportunities exist to improve **unsupervised models**, which currently struggle with reliability in educational contexts.

13. Hard ML Quotes / Definitions

• "Machine Learning is a set of techniques that gives computers the ability to learn without the intervention of human programming".

• "The goal of SL [Supervised Learning] is to build a clear model of the distribution of class labels in terms of predictor characteristics".

• "Success has been more in recommending resources and activities than in predicting student behavior" [regarding Recommender Systems].

14. Tags / Keywords

["student_performance_prediction", "supervised_learning", "dropout_detection", "SVM", "recommender_systems", "educational_data_mining", "learning_analytics"]

15. Relevance Score & Subtype

- **ML Relevance:** 5/5
- **Educational Outcome Relevance:** 5/5
- **Applicability to Auri's Journey:** 5/5
- **Novelty for Thesis:** 4/5

16. Notes for Thesis Synthesis Section

Rastrollo-Guerrero et al. (2020) establish that **Supervised Learning**—particularly **Support Vector Machines (SVM)**—remains the gold standard for predicting student performance and dropout risk, with some models achieving over 97% accuracy. Their review emphasizes that while university-level data is abundant, there is a critical need for early-stage predictive tools in **primary and secondary education**, and that integrating **non-cognitive behavioral data** into these models significantly enhances their pedagogical utility.

1. Citation

Fahad Mon, B., Wasfi, A., Hayajneh, M., Slim, A., & Abu Ali, N. (2023). "Reinforcement Learning in Education: A Literature Review." *Informatics*, 10(3), 74.

2. Study Aim & Context

• **Main Goal:** To investigate the applications and techniques of **Reinforcement Learning (RL)** in education, determine its potential for enhancing outcomes, and identify best practices for its incorporation into educational settings.

• **Focus:**

◦ **Intelligent Tutoring Systems:** Developing automated teaching strategies and simulating human student behavior.

◦ **Personalized/Adaptive Learning:** Adjusting difficulty levels and instructional sequencing based on individual performance.

◦ **Curriculum Design:** Restructuring STEM programs based on course complexity and graduation rates.

• **Field:** Computer and Network Engineering, Artificial Intelligence, and Educational Technology.

3. Core ML Problem & Purpose

The sources define RL as a computational paradigm for **sequential decision making**. The primary ML purpose in this context is to:

• **Optimize long-term utility:** Training agents to map states to actions (policies) that maximize future educational rewards, such as mastery or reduced learning time.

• **Model hidden states:** Using frameworks to handle factors that are not directly observed, such as a student's actual proficiency or cognitive grasp.

• **Automate decision making:** Removing the need for manual, hand-crafted instructional scripts by allowing the agent to learn through direct contact with the learning environment.

4. Methodology & ML Approach

Aspect	Details

Research Type	**Systematic Literature Review** following the PRISMA methodology.
Target Learners	Multi-level, ranging from **school students** (arithmetic/physics) to **higher education** (engineering/algebra).
Data Used	Interaction experience, log files, historical student data, university records, and simulated student models.
ML Tasks	**Reinforcement Learning** (Sequential decision making and goal-directed learning).
Algorithms	**Markov Decision Process (MDP)**, **POMDP**, **Deep RL**, and **Markov Chains**.
Training Process	Synthesis of existing studies using online/offline learning, simulated training phases, and evaluation against baselines.

5. Model Input Features

The sources detail a wide range of features used to train educational RL models:

• **Academic/Performance:** Correct/incorrect responses, normalized learning gain (NLG), points accumulation, and letter grades.

• **Engagement/Behavior:** Time spent on problems, hint requests, and student "thoughts" or learning situation descriptions.

• **Latent States:** Belief states space representing unobserved student proficiency and knowledge.

• **Curricular Structure:** Prerequisites, **blocking factors** (number of classes restricted until completion), and **delay factors** (number of prerequisite pathways).

• **Socio-Demographics:** Family assistance, race/sex, and pre-institutional preparation.

6. Model Output & Interpretation

• **Model Prediction/Output:**

  ◦ **Optimal Teaching Policy:** Decisions on what action to take next (e.g., provide a hint, show a worked example, or move to a new topic).

  ◦ **Knowledge State:** Estimated probability of mastery for specific knowledge components.

  ◦ **Next Best Resource:** Recommendations for learning objects or tailored assignments.

• **Usage within System:**

◦ **Triggering Adaptive Sequencing:** Dynamically ordering activities to optimize the learning path.

◦ **Teacher Advising:** Guiding student exploration through an advising mechanism (Teacher-Student framework).

7. Algorithms / Techniques Compared

The sources provide a benchmark for four distinct RL techniques.

Algorithm	Pros Reported	Cons Reported	Best Use Case
MDP	Improved management/planning by finding the most profitable sequential decisions.	Assumes full observability of student states, which is rarely true.	Curricular analytics and instructional sequencing.
POMDP	Handles **latent knowledge** and unobserved factors via belief states.	Challenging to solve; often requires approximate methods or myopic planning.	Cognitive mastery learning and latent knowledge tracing.
Deep RL	Uses neural networks to handle complex, high-dimensional tasks.	Requires significant data and computational resources.	Personalized interactive narratives in educational games.
Markov Chain	Quantitative analytic technique based on stochastic processes; good for teaching quality evaluation.	Primarily used for evaluation rather than active, goal-directed intervention.	Evaluating classroom teaching ability and blended learning.

8. Key Findings & Insights

• **Effectiveness:** RL is an effective framework for enhancing personalized outcomes, often outperforming hand-crafted or random policies in speed and learning gain.

• **Efficiency:** New RL-induced policies in systems like AnimalWatch and RLATES significantly reduced the time students spent on problems while maintaining or improving knowledge levels.

• **Exploration-Exploitation:** A critical but rarely addressed trade-off in education is balancing the best known policy (exploitation) with trying new, uncertain actions (exploration).

• **Incentive Alignment:** RL teachers can adapt to when and how to provide advice based on the student's state and budget.

9. Pedagogical / Learning Implications

• **Personalization:** RL shifts teaching from "one size fits all" to adaptive difficulty that adjusts based on real-time performance.

• **Teacher Workload:** Adaptive scheduling and automated hint generation lighten the workload for instructors and course designers.

• **Active Learning:** RL facilitates goal-directed exploration, allowing students to learn through feedback rather than just following instructions.

• **Transdisciplinarity:** The sources argue that AI is not just a STEM field but necessitates skills in humanities and social sciences to address ethical and socio-emotional factors.

10. Practical Design Implications for Auri's Journey

• **Hybrid Modeling:** Combine **data-driven RL** with **psychological theories** to guide the selection of models and strategies.

• **Expertise Reversal Effect:** Design the system to start with worked examples and gradually reduce them in favor of problem-solving tasks as mastery increases.

• **Limit Choices:** Focus on scenarios with **limited yet meaningful choices** to simplify the state-action space for the RL agent.

• **Trajectory Tracking:** Utilize **Advice Replay Memory (ARM)** to allow students to effectively reuse advice provided by the system.

11. Strengths & Novel Contributions

• **Broad Framework:** Provides a universal framework for AI and automated decision-making across diverse educational contexts.

• **Curricular Analytics:** Introduces a novel approach for quantifying the relationship between curriculum complexity and graduation rates using MDP.

• **Teacher-Student Interaction:** Formalizes the teacher-student advising mechanism to improve sample efficiency.

12. Limitations & Identified Gaps

• **Understudied Socio-Emotional Factors:** Most studies focus on cognitive results, ignoring the emotional impact of AI on students.

• **Data Selection Ethics:** Efficient predictive models require unstructured data, which raises significant **privacy and ethical concerns**.

• **Expertise Gap:** Teachers often lack the technical expertise to understand "black box" AI recommendations.

• **Resource Scarcity:** A lack of standardized, reusable digital "learning objects" limits the ability to customize adaptive platforms effectively.

13. Hard ML Quotes / Definitions

• "**Reinforcement Learning (RL)** can be defined as a computational paradigm for sequential decision making and goal-directed learning".

• "The **Exploration-Exploitation dilemma**: Exploration involves attempting new actions... while exploitation entails utilizing the best policy identified thus far".

14. Tags / Keywords

["reinforcement_learning", "Markov_decision_process", "instructional_sequencing", "adaptive_experimentation", "curricular_analytics", "personalized_learning", "POMDP"]

15. Relevance Score & Subtype

• **ML Relevance:** 5/5

• **Educational Outcome Relevance:** 5/5

• **Applicability to Auri's Journey:** 5/5

• **Novelty for Thesis:** 5/5

16. Notes for Thesis Synthesis Section

Fahad Mon et al. (2023) provide a comprehensive review of **Reinforcement Learning (RL)** as a framework for sequential decision-making in education, demonstrating its superiority over hand-crafted policies in personalizing learning paths and instructional sequencing. Their work emphasizes the importance of **hybridizing data-driven models with psychological principles** and highlights **curricular analytics** as a critical application for using Markov Decision Processes to optimize academic progress and institutional graduation rates.

1. Citation

Lin, C. C., Huang, A. Y. Q., & Lu, O. H. T. (2023). "Artificial intelligence in intelligent tutoring systems toward sustainable education: a systematic review." *Smart Learning Environments*, 10(41).

2. Study Aim & Context

• **Main Goal:** To bridge the perspectives of educators and IT specialists to connect education with AI technology, specifically reviewing how AI-embedded Intelligent Tutoring Systems (ITS) support sustainable education.

• **Focus:**

◦ Intelligent Tutoring Systems (ITS)

◦ Personalized and adaptive learning experiences

◦ Performance prediction and real-time monitoring

◦ Sustainable education (addressing SDGs)

• **Field:** Computer Science, Educational Technology, and Information Technology.

3. Core ML Problem & Purpose

The research addresses the challenges of providing **personalized, high-quality education** at scale. The ML purpose includes:

• **Predicting student mastery/performance:** Using historical data to forecast outcomes.

• **Modeling learning behavior:** Categorizing how students interact with systems (clustering).

• **Detecting engagement/dropout:** Using behavioral and emotional cues to identify at-risk students.

• **Personalizing task difficulty/path:** Creating adaptive learning sequences.

• **Classifying psychological states:** Recognizing emotions (facial expressions/gestures) to inform teacher interventions.

4. Methodology & ML Approach

Aspect	Details
Research Type	**Systematic Review** using PRISMA methodology.
Target Learners	**University** (19 papers), High School (4 papers), Elementary, and K-12.
Data Used	System log files, interaction data (clickstreams), academic data (grades), and facial biological features.
ML Tasks	**Classification, Regression, Clustering,** and Natural Language Processing (NLP).
Algorithms	**Supervised:** SVM, RF, NB, NN, Logistic Regression, XGBoost, CART. **Unsupervised:** KNN (for behavior clustering), HMM, CNN (for facial recognition), and NLP (DistilBERT, LDA).
Training Process	Synthesis of papers using various evaluation metrics (Accuracy, verified through experimental school scenarios).

5. Model Input Features

- **Academic/Performance Data:** Grades, quiz scores, and prior knowledge.
- **Interaction/Behavioral Data:** Clicks, system logs, mind map usage, and time-on-task.
- **Emotional/Biometric Data:** Facial expressions, gestures, and emotions detected via camera.
- **Environmental Context:** Network connectivity (bandwidth) and device type.

6. Model Output & Interpretation

- **Model Prediction:**
  - **Performance outcome:** Likely grade or success probability.
  - **Behavioral clusters:** Categories of learning styles.
  - **Dropout risk:** Notifications for early intervention.
- **Usage within System:**
  - **Triggering adaptive hints/scaffolding:** Real-time assistance during tasks.
  - **Informing Instructors:** Providing dashboards showing student emotions or engagement.

◦ **Adjusting learning paths:** Dynamically selecting the next best learning resource.

7. Algorithms / Techniques Compared

The review categorizes performance based on the specific educational objective.

Algorithm	Pros Reported	Cons Reported	Best Use Case
Supervised Learning (SVM, RF)	High accuracy for performance prediction.	Risk of **bias and overfitting**; requires labeled data.	Grade/dropout prediction.
Unsupervised (KNN, Clustering)	Good for identifying **behaviors** without ground truth data.	Harder to verify accuracy without labels.	Categorizing learning patterns.
NLP (Chatbots/BERT)	Interactive; provides immediate feedback/intervention.	Can be difficult to integrate into specific curricula.	Feedback/Intervention.
XAI (SHAP, Counterfactual)	Increases **trust and transparency** for educators.	Still early-stage; increases technical complexity.	Explainable predictions.

8. Key Findings & Insights

• **Personalization Boost:** AI-embedded systems offer better adaptation to real-time student status than non-embedded systems.

• **Teacher Efficiency:** ITS systems offload routine tasks like grading, allowing teachers to focus on student-centered instruction.

• **Performance over Explanation:** While accuracy is often high (10 studies verified this), most research fails to explain **how** the AI arrived at the prediction (the "black box" problem).

• **Sustainable Impact:** AI supports **SDG Goal 4 (Quality Education)** by bridging distance gaps and providing equal access regardless of location.

9. Pedagogical / Learning Implications

• **Student-Centeredness:** Shifts the paradigm from teacher-centered to student-centered by providing individualized learning trajectories.

• **Metacognitive Support:** Feedback and scaffolding help students develop **lifelong learning** and self-regulation skills.

• **Equitable Access:** Remote and hybrid learning empowered by AI ensures inclusive education for diverse learners.

10. Practical Design Implications for Auri's Journey

• **Implement Explainable AI (XAI):** Incorporate an **explanation interface** (like SHAP) so users (teachers/students) understand *why* the system recommends a specific path.

• **Feature Diversity:** Use **facial/emotional recognition** alongside log data to get a holistic view of learner engagement.

• **Curriculum Integration:** Ensure the AI is **tightly embedded** into the learning process rather than just analyzing data "offline" to allow for real-time interventions.

• **Policy Compliance:** Address **privacy and bias** early in the design phase to align with government regulations.

11. Strengths & Novel Contributions

• **Holistic Stakeholder View:** Combines insights from both **IT architects and educators**.

• **Sustainability Focus:** Directly links AI implementation to the **United Nations' Sustainable Development Goals (SDGs)**.

• **Standardized Framework:** Uses **PRISMA** and the analytical model from Chatti et al. for rigorous review.

12. Limitations & Identified Gaps

• **Explainability Gap:** Most current models are "black boxes," making it hard for educators to trust the interventions.

• **Data Portability:** AI solutions developed for one tutoring system are often not easily **portable** to others.

• **Human-Centered Design:** There is a need for more **Human-Centered AI (HCAI)** that focuses on socio-emotional factors and reliability/safety.

• **Under-representation of Lower Ed:** Most research is heavily skewed toward **higher education (University)**.

13. Hard ML Quotes / Definitions

• "**Sustainable education** aims to foster a learning culture that values diversity, creativity, and participation and empowers learners to develop sustainably."

• "**Adaptive learning** is a personalized approach to learning that makes the best out of every student's learning."

14. Tags / Keywords

["intelligent_tutoring_system", "sustainable_education", "learning_analytics", "performance_prediction", "XAI", "personalized_learning", "human_centered_AI"]

15. Relevance Score & Subtype

• **ML Relevance:** 5/5

• **Educational Outcome Relevance:** 5/5

• **Applicability to Auri's Journey:** 5/5

• **Novelty for Thesis:** 4/5

16. Notes for Thesis Synthesis Section

Lin et al. (2023) demonstrate that **AI-embedded Intelligent Tutoring Systems** are critical to achieving **sustainable education goals** by providing personalized learning at scale and reducing teacher workload on routine tasks. While supervised learning models like **SVM and Neural Networks** achieve high predictive accuracy for student performance, the authors emphasize that the future of educational AI must shift toward **Human-Centered AI (HCAI) and Explainable AI (XAI)** to ensure fairness, transparency, and trust within the classroom environment.