AI IN PERSONALIZED LEARNING

INTRODUCTION

Traditional digital learning platforms often employ a "one-size-fits-all" methodology, presenting the same educational content to all students regardless of their individual learning pace, behavioral patterns, or engagement levels. This static approach can fail to meet the specific needs of diverse learners, potentially leading to disengagement for advanced students or frustration for those who are struggling. This document outlines an AI-powered system designed to overcome these limitations. By leveraging machine learning models to analyze real-time student interaction data, the system provides a dynamic, personalized, and adaptive learning experience. It achieves this by classifying learner types, predicting academic performance, and analyzing engagement to tailor content and support for each unique student.

1. DATASET: ASSISTMENTS EDUCATIONAL DATA

The foundation of this project is the ASSISTments dataset, a comprehensive collection of student interaction data from an online mathematics tutoring platform. This rich, real-world dataset is ideal for this application as it contains fine-grained records of the learning process, including:

- **Student Answers:** Correctness of each response.
- **Temporal Data:** Time taken to answer each question.
- **Interaction Metrics:** Number of attempts and hints used per problem.
- **Problem Metadata:** Information about the questions themselves, such as topic and difficulty.

By capturing the nuances of how students approach and solve problems, the ASSISTments data provides the necessary features to build robust models that can understand and predict learning behaviors. For this prototype, a representative sample of the data was used to ensure rapid development and iteration.

2. SYSTEM ARCHITECTURE: CORE AI MODELS

The system is built upon three distinct but interconnected machine learning models, each serving a specific function in creating a personalized learning path.

2.1. Learner classification model

- **Purpose:** To classify students into distinct learning archetypes based on their overall behavior and performance patterns. This allows the system to understand a student's general approach to learning. The defined learner types are: **Advanced, Moderate, Struggling, and Balanced**.
- **Algorithm:** Random Forest Classifier.
- **Features Analyzed:** The model is trained on a set of engineered features that summarize a student's session:
    - accuracy: Overall percentage of correct answers.
    - total_questions: Total number of questions attempted.
    - avg_time_seconds: The average time spent per question.
    - avg_attempts: The average number of attempts made per question.
    - avg_hints_used: A measure of help-seeking behavior.
    - consistency: The standard deviation of performance, indicating stability.
    - speed_accuracy_tradeoff: A metric balancing how quickly a student answers versus their correctness.
    - persistence: A measure of a student's willingness to retry after an incorrect answer.
    - engagement: A composite score reflecting active participation.
    - efficiency: A metric combining accuracy, speed, and attempts.
- **Model Specifications:** n_estimators: 200, max_depth: 10.

2.2. Performance prediction model

- **Purpose:** To predict the probability of a student successfully answering the *next* question they face. This real-time forecast enables the system to dynamically adjust the difficulty of upcoming questions, preventing frustration or boredom.
- **Algorithm:** Gradient Boosting Classifier.
- **Features Analyzed:** This model focuses on features from the most recent interactions:
    - attempts: The number of attempts on the current or previous question.
    - time_taken_seconds: The time spent on the current or previous question.
    - hints_used: The number of hints requested for the current or previous question.
    - attempt_efficiency: An efficiency metric calculated as 1 / attempts.
- **Model Specifications:** n_estimators: 100, max_depth: 10.

2.3. Engagement Analysis Model

- **Purpose:** To assess a student's level of engagement in real-time, categorized as **Low, Medium, or High**. This allows the system to identify signs of waning motivation and proactively offer encouragement or alternative activities.
- **Algorithm:** Random Forest Classifier.
- **Features Analyzed:** The model evaluates engagement based on a student's recent pattern of interaction:
    - total_interactions: The number of questions attempted in the current session.
    - avg_accuracy: The student's average performance in the session.
    - accuracy_std: The consistency of the student's performance.
    - avg_attempts: The average attempts per question in the session.
    - avg_time: The average time per question in the session.
- **Model Specifications:** n_estimators: 150, max_depth: 10.

3. RATIONALE FOR MODEL SELECTION (Considering a Hybrid System)

The choice of **Random Forest** and **Gradient Boosting** was deliberate and based on the nature of the data and the project goals.

- **Random Forest (Learner Classification & Engagement):** This algorithm is highly effective for classification tasks with tabular data. It is robust to outliers, handles a mix of feature types without extensive pre-processing, and inherently provides feature importance metrics, which aids in model interpretability. Its ensemble nature also protects against overfitting, which is crucial when working with smaller, prototype-scale datasets.
- **Gradient Boosting (Performance Prediction):** This model excels at capturing complex patterns and delivering high predictive accuracy, making it ideal for the nuanced task of predicting future performance. It builds trees sequentially, with each new tree correcting the errors of the previous one, a process that is well-suited for modeling sequential student behavior.
- **Alternatives Considered:** More complex models like Neural Networks were deemed less suitable for this prototype stage due to their need for larger datasets, extensive hyperparameter tuning, and their "black-box" nature, which complicates interpretability. Similarly, Support Vector Machines (SVMs) can be less efficient on larger datasets and more complex to tune for multi-class problems.

4. TRAINING AND IMPLEMENTATION

Training Protocol

The models were trained following a standard machine learning pipeline. After loading and pre-processing the data from the ASSISTments sample, features were engineered, and any missing values were imputed. The data was then split, with 80% used for training and 20% reserved for testing. A **5-fold cross-validation** strategy was employed during training to ensure the models' robustness and generalizability.

It is important to note that as tree-based ensemble methods, Random Forest and Gradient Boosting do not utilize gradient descent in the same manner as neural networks. Therefore, the concepts of **epochs and batch sizes are not applicable**. The models are trained on the entire training dataset at once. A random_state of 42 was used throughout to ensure full reproducibility of the results.

Real-Time Prediction Pipeline

In the live application, the models operate in a continuous feedback loop:

1. **Data Ingestion:** A student begins a quiz, and their interactions (answers, time, attempts, hints) are captured.
2. **Feature Extraction:** The system calculates the relevant features for each of the three models in real-time.
3. **Model Prediction:** The feature vectors are passed to the trained models, which generate simultaneous predictions for learner type, performance probability, and engagement level.
4. **Adaptive Response:** Based on the model outputs, the system generates a personalized learning plan, which may include adjusting the difficulty of the next question, providing motivational messages, or suggesting supplementary material.
5. **Display:** The recommendations and adaptive content are presented to the student.

5. CONCLUSION AND FUTURE ENHANCEMENTS

This project successfully demonstrates a prototype for an AI-driven adaptive learning system. By using interpretable and effective machine learning models, it provides a solid foundation for creating truly personalized educational experiences. The system can identify student needs, predict performance, and track engagement, moving beyond the static limitations of traditional e-learning platforms.

Potential future enhancements include:

- **Scaling:** Training the models on the full ASSISTments dataset to improve accuracy and capture more nuanced learning patterns.
- **Model Exploration:** Experimenting with deep learning models, such as LSTMs or Transformers, to better capture the sequential nature of learning data.
- **Advanced Personalization:** Expanding the recommendation engine to support multiple subjects and a wider variety of content types.

This work represents a meaningful step toward building more effective, engaging, and equitable digital learning environments.

From

Kadali Harshavardhan

Project: AI in Personlized Learning

IIT Ropar