Decoding Data Science(DDS) Academy

AI Application Building Challenge

Day 1 Idea Submission Template

Project Title:

AI-Driven System for Student Outcomes Assessment and Continuous Improvement

Concept Summary:

AI-Driven System for Student Outcomes Assessment and Continuous Improvement integrates artificial intelligence to streamline the evaluation of student outcomes (SOs) in educational institutions. By automating data analysis, applying rule-based decision-making, and generating actionable insights, the system ensures accurate, consistent, and transparent assessments. It not only evaluates SOs but also identifies underperforming areas, providing targeted improvement recommendations to enhance curriculum design and academic performance.

Target Audience:

Educational Administrators: Seeking data-driven tools for curriculum assessment and improvement.

Instructors: Aiming for consistent and objective evaluation of student outcomes.

Students: Benefiting indirectly through targeted curriculum enhancements and improved learning experiences.

Key Features:

Rule-Based Expert System: Uses predefined logic to evaluate Performance Indicators (PIs) and student outcomes.

Automated Data Processing: Normalizes and analyzes assessment data to generate meaningful insights.

Targeted Recommendations: Offers specific plans for underperforming PIs and SOs to drive improvement.

Dynamic Reports: Generates detailed achievement summaries with categorized performance levels (e.g., Needs Improvement, Satisfactory, Excellent).

Scalability: Applicable across programs, campuses, and varying academic structures.

Technical Approach:

1. Knowledge Base: Stores rules and facts for evaluating PIs and SOs, including predefined scoring thresholds, rubrics, and performance benchmarks.

2. Inference Engine: Employ forward-chaining reasoning to assess outcomes and generate targeted recommendations for improvement.

3. KNIME Integration: The system uses KNIME as a no-code/low-code platform for advanced data workflows, including preprocessing assessment data, implementing rule-based logic, and generating predictive analytics for continuous improvement.

4. Database: PostgreSQL for structured data storage, enabling efficient queries and analytics.

5. Visualization: KNIME's built-in visualization capabilities, along with tools like Matplotlib and Power BI, are used to create insightful dashboards and reports that highlight areas of excellence and improvement.

Expected Challenges:

Data Integration: Difficulty in gathering standardized assessment data from diverse systems.

- **Solution**: Develop APIs and data mapping techniques for seamless integration.

AI Transparency: Gaining trust in AI-driven evaluations.

- **Solution**: Incorporate explainable AI methods to clarify decision-making processes.

Submission Format:

- Detailed project proposal outlining the system's design and implementation.
- A presentation showcasing the problem, solution, and impact.
- Interactive demo with sample data to illustrate the system's capabilities.

Expected Outcome:

- A functional AI-driven prototype capable of assessing student outcomes and generating actionable recommendations.
- Improved efficiency and accuracy in evaluating educational programs.
- A roadmap for integrating AI in continuous academic improvement processes.

Additional Notes (Optional):

Future extensions could include real-time integration with LMS platforms and predictive analytics for student performance.

Day2

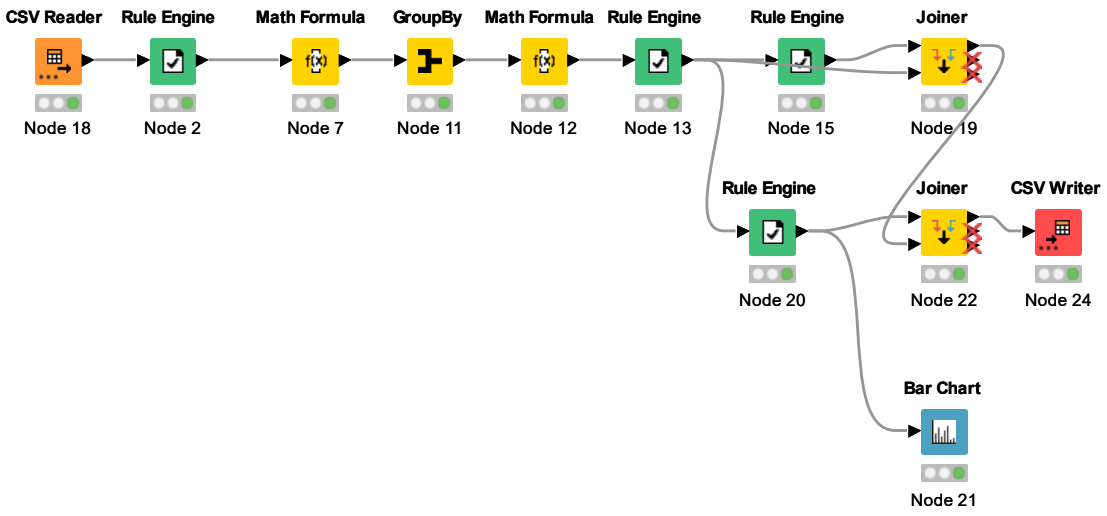
Environment Setup

1. Infrastructure Preparation:
    - Server/Cloud: Set up a cloud environment (e.g., AWS, Azure, or GCP) or a local server with sufficient storage and compute power for hosting KNIME and PostgreSQL.
    - Database Setup: Install and configure PostgreSQL for storing assessment data, rubrics, and benchmarks.
    - Development Tools:
        - Install KNIME for workflow creation and data processing.
        - Set up Python for advanced analytics and integration with Matplotlib or other libraries.
        - Install Power BI or Tableau for visualization needs.
2. Integration Framework:
    - Develop APIs to enable seamless data transfer between the system, LMS platforms, and other academic tools.
    - Define data mapping techniques to standardize diverse assessment data formats.

Initial Development

1. Data Collection and Preparation:
    - Gather sample assessment data for Performance Indicators (PIs) and Student Outcomes (SOs).
    - Preprocess data using KNIME, ensuring it is clean, normalized, and ready for analysis.
2. Knowledge Base Creation:
    - Define the rules and thresholds for evaluating PIs and SOs.
    - Store these rules in the knowledge base within KNIME or PostgreSQL.

3. Inference Engine:
    - Build forward-chaining logic in KNIME to evaluate PIs and generate recommendations.
    - Test the engine with sample data to validate its decision-making capabilities.
4. Visualization and Reporting:
    - Develop interactive dashboards and reports using KNIME and Power BI.
    - Include categorized performance levels (Needs Improvement, Satisfactory, Excellent) to provide actionable insights.
5. Prototype Development:
    - Create a functional prototype integrating all components: data processing, rule-based evaluations, and reporting.
    - Ensure the system is modular for scalability and future extensions.



Dataset

1. **Source**: Assessment data from educational institutions.
2. **Structure**:

- Performance Indicators (PIs) mapped to Student Outcomes (SOs).
- Predefined scoring thresholds and rubrics.
- Student performance data categorized by academic programs, courses, and individual achievements.

3. **Challenges**:
    - Standardization of data formats from diverse systems.
    - Integration of historical and current datasets.