

# HW2 - “Analyst Track” Comparative Analysis of AI Architectures for Adaptive Language Assessment

**Course:** MIT MAS.665 - AI for Impact

**Author:** Rong Wu

**Date:** September 24, 2025

---

## 1. TL-DR & Comments

- a. Read section (2) and (6) if you just want to get the summary.
  - b. I have hosted my app on AWS server and it seems like I was stopped by a bug of the speaking module that works locally (but not on the AWS-server <http://54.83.88.109:8000/app/speaking/>)
  - c. Thank you for this course!
- 

## 2. Introduction

The field of artificial intelligence offers multiple architectural paradigms for building intelligent systems. The choice of architecture be it a classic rule-based system, a predictive machine learning model, or a modern autonomous agent carries significant trade-offs in adaptability, scalability, explainability, and autonomy. This report analyzes these three approaches through the practical lens of the Adaptive English Placement Agent, a project developed for this course and because I come from education background.

The project is a web application designed to assess a user’s English language proficiency through a series of adaptive modules (Reading, Writing, Speaking, etc.). Similar to testing the level of English (A1-C2). It leverages Google’s Gemini large language model (LLM) and Grok (Openroute) as a fallback (because Gemini, even paid, is so busy!!) to generate questions, evaluate user responses in real-time, and adjust the difficulty level based on performance. This implementation serves as our primary example of an agent-based system.

This document will first describe the implemented agent-based architecture and its characteristics. It will then explore how the same application could have been constructed using a rule-based system and traditional machine learning models. Finally, it will offer a comparative analysis to determine the most practical approach for the educational technology sector today.

---

### 3. The Implemented Solution: An Agent-Based System

The Adaptive English Placement Agent is an AI agent that performs the complex task of language assessment. It can be defined as an agent because it perceives its environment (user inputs), processes that information, and acts autonomously to achieve specific goals (accurately assessing proficiency).

- **Architecture:** The system uses a FastAPI backend to connect a web-based frontend to the LLM. When a user answers a question, the agent sends the prompt, context, and user’s answer to the LLM API. It then interprets the model’s response (e.g., a score, a critique, or a new, more challenging question) and updates the user’s state and interface accordingly.
- **Benefits & Limitations:**
  - **Adaptability (High):** This is the architecture’s greatest strength. The agent can generate a virtually infinite variety of questions and understand nuanced, free-form text and speech. It is not limited to a pre-defined question bank, allowing it to probe a user’s skill in novel ways.
  - **Scalability (High):** The system’s core logic scales effortlessly, as the heavy lifting of content generation and evaluation is handled by the highly scalable LLMs. The application backend primarily manages state and API calls. Cool bonus: we can run few in parallel & because it’s mostly text it’s not expensive. Super useful for caching (killing long latency).
  - **Explainability (Low):** The reasoning behind an LLM’s evaluation can be opaque. While it can be prompted to “explain its reasoning,” the internal decision-making process is a black box, which can be a drawback in high-stakes assessment scenarios where transparent scoring rubrics are required. Basically it’s not fully deterministic.
  - **Autonomy (High):** The agent operates with a high degree of autonomy. It independently generates questions, evaluates responses, and decides on the next level of difficulty without human intervention or reliance on a static decision tree.

### 4. Alternative Approach 1: A Rule-Based System

A classical AI approach would involve constructing the assessment tool from a set of explicit, hard-coded rules.

- **Architecture:** The system would rely on a large, structured database of questions, each tagged with a specific skill and difficulty level (e.g., A1, B2, C1). The logic would be a massive IF-THEN-ELSE decision tree.
  - IF a user answers 8 out of 10 questions correctly at level B1, THEN present them with B2-level questions.
  - For writing evaluation, it would use regular expressions and keyword

matching: IF the answer CONTAINS “have been” and NOT CONTAINS “has been”, THEN mark grammar point X as correct.

- **Benefits & Limitations:**

- **Adaptability (Very Low):** The system is rigid. It cannot understand or correctly score answers that deviate slightly from the expected patterns. It cannot generate new content, making the experience repetitive and predictable. It breaks down completely when faced with the complexity and creativity of human language.
- **Scalability (Poor):** While the system could handle many users, scaling the *content* is a monumental, manual task. Every new question, every rule, and every possible correct answer variation must be programmed by hand.
- **Explainability (Very High):** The system is perfectly transparent. Every decision and score can be traced back to a specific, human-written rule. This is its primary advantage.
- **Autonomy (None):** The system has no autonomy; it is a deterministic machine executing a pre-defined script.
- **Other:** Huge con is that for smaller DB the user can just learn the questions and take a wised guess.

A rule-based system is only sufficient for the most basic, multiple-choice style quizzes where answers are unambiguous. It fails to provide a meaningful assessment of generative language skills like writing or speaking.

## 5. Alternative Approach 2: A Machine Learning Model

A more modern, non-agentic approach would use traditional machine learning (ML) models trained for specific predictive tasks.

- **Architecture:** This would not be a single system but a collection of specialized models.
  1. **Question Selector:** An algorithm (like a recommender system) that selects the next best question from a large, pre-existing database based on the user’s performance history.
  2. **Proficiency Classifier:** A classification or regression model trained on thousands of sample essays and their corresponding scores to predict a user’s proficiency level (e.g., B2) from their written input.
  3. **Speech-to-Text + Grader:** A pipeline that first transcribes spoken audio and then feeds the text into the proficiency classifier.
- **Benefits & Limitations:**
  - **Adaptability (Moderate):** This approach is more adaptable than a rule-based system. It can identify complex patterns in language that rules cannot capture. However, it is still fundamentally limited to the knowledge contained within its training data and its pre-defined question bank. It cannot generate novel content.
  - **Scalability (Good):** Once trained, ML models can serve predictions to thousands of users efficiently. However, the initial data collection

and training process is expensive and time-consuming.

- **Explainability (Low to Moderate):** While less of a black box than massive LLMs, explaining a specific prediction from a deep learning model is still challenging. Techniques like LIME or SHAP can provide insights but do not offer the simple clarity of a rule.
- **Autonomy (Low):** The models perform predictive tasks but have low autonomy. They select, classify, or score, but they do not generate, reason, or strategize in the way an agent does.
- **Other:** I have used OCR in my app to scan the text out of a written page, but the agent actually makes the decision about the score of the content.

ML models offer a significant improvement over rule-based systems for handling nuance but come with the high overhead of data sourcing and training, and they lack the generative and conversational power of agents.

## 6. Conclusion & Sector-Specific Recommendation

The choice of architecture fundamentally defines the capabilities and limitations of an AI assessment tool.

Feature	Rule-Based System	Machine Learning Model	Agent-Based System (Implemented)
<b>Adaptability</b>	Very Low	Moderate	<b>High</b>
<b>Scalability</b>	Poor (Content)	Good (Users)	<b>High</b>
<b>Explainability</b>	Very High	Low to Moderate	Low
<b>Autonomy</b>	None	Low	<b>High</b>

For the educational technology sector, particularly in language learning, the agent-based architecture is the most practical and powerful approach today. Its core strength in generating novel, context-aware content and evaluating free-form human input aligns directly with the needs of meaningful language assessment. While rule-based systems are too brittle and traditional ML models are too constrained, LLM-powered agents provide the dynamic, interactive experience necessary to accurately gauge a learner’s true capabilities.

The primary challenge for agentslow explainabilityis a significant consideration. A hybrid approach may be the ultimate solution: using an agent for the dynamic, conversational parts of the assessment while employing rule-based checks or specialized ML models for scoring specific, objective criteria (e.g., counting grammatical errors of a specific type). However, for creating a truly adaptive and comprehensive placement test, the agent-based paradigm represents a clear and decisive step forward.