

Universiteit Leiden

# Master Computer Science

Learning to Prompt: Improving Student Engagement with Adaptive LLM-based High-School Tutoring


| | |
|---|---|
| Name: | Po-Chin Chang |
| Student ID: | 4272455 |
| Date: | 26/06/2026 |
| Specialisation: | Artificial Intelligence |
| 1st supervisor: | Aske Plaat |
| 2nd supervisor: | Michiel T. van der Meer |


Master's Thesis in Computer Science


Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

# Abstract

This thesis presents a Subject-aware Prompt Routing framework designed to optimize pedagogical strategy selection in LLM-powered tutoring systems. While Large Language Models (LLMs) offer potential for personalized education, static prompting often fails to adapt to diverse academic contexts and student needs. To address this, we developed a feature-based score mechanism that quantifies instructional quality by extracting 14 pedagogical dimensions (e.g., tutor scaffolding, student curiosity) from raw conversation transcripts. This provides an immediate signal to the prompt router, avoiding the sparsity of exercise scores from real students. Additionally, by integrating expert consensus and applying a calibrated score smoothness coefficient, we aligned the synthetic training distribution with real-world pedagogical expectations.

Our framework utilizes a Subject-aware Prompt Routing model pretrained in a simulation environment and subsequently deployed for continuous online adaptation. Empirical results from real-world deployment demonstrate a sim-to-real transition. The routing model unlearned simulation-induced biases and adapted to more effective scaffolding techniques (e.g., the *Coach* strategy). Statistical analysis from 359 real Dutch high school students confirms that the adaptive prompting improves instructional efficiency in interactions ($p = 0.007$) by around 3 turns. Furthermore, our analysis proves that higher pedagogical scores significantly predict student engagement ($p = 0.037$). Exercise conversion is improved by the router, achieving a 1.2% increase (from 19.6% to 20.8%).

This research underscores that educational outcomes can be systematically quantified beyond raw test scores. By dynamically adapting pedagogical strategies across diverse subjects, the router effectively enhances the overall learning experience for real-world students. Overall, this framework provides a foundation for adaptive LLM-based tutoring systems in the education field. Future work will consider higher-capacity models, more authentic student simulations, and deeper system integration.

# Acknowledgements

I would first like to thank my supervisor, Professor Aske Plaat. His lively lecture style originally inspired me to begin my journey in the field of Reinforcement Learning. Under his guidance and encouragement, I was able to stay focused on my research questions and successfully develop this thesis. His insightful feedback was always helpful and opened up great opportunities for publishing this work.

I am also deeply grateful to my second reader, Michiel. While Michiel participated in the process later during the thesis writing, his domain knowledge in Large Language Models was invaluable. His input is critical for completing and refining this thesis.

Special thanks to the company FutureWhiz for giving me this opportunity to work on Artificial Intelligence in the education field. My company supervisor, Nick, was always passionate and supportive, and I learned many practical considerations and educational domain insights from him. Also, all members of the AI team are very helpful. The education experts, Jasper and Renske, provided me with a wealth of knowledge about data science in education. The AI engineers, Sumesh and Dung, gave me a strong foundation for implementing the AI routing algorithm into a production environment.

Finally, I would express my deepest gratitude to my family and my wife, Demi. Thanks for your love, encouragement, and endless support, especially during the most stressful moments of preparing this thesis. This milestone belongs to you as much as it does to me.

# Ethics Statement

This thesis involves the collection and analysis of real conversations from high school students. All student conversation transcripts were processed within a secure, closed-loop environment by a professional high-school tutoring company, Futurewhiz. Data were handled following the General Data Protection Regulation (GDPR), and access was restricted to the research team. Explicit consent for data usage for research purposes was obtained through agreements with the participating educational platform.

To ensure AI safety and child protection, the Large Language Model (LLM) used in our tutoring system was constrained to educational content only. Strict system prompts and guidelines ensure that all conversions avoid violent, explicit, or otherwise inappropriate and controversial topics.

Lastly, the goal of this research is to enhance the learning quality and student engagement by an LLM-based tutoring system, while maintaining the highest standards of student safety, privacy, and well-being.

# Contents

**Abstract** i

**Acknowledgements** iii

**Ethics Statement** iv

**1 Introduction** 1
- 1.1 Problem Statement . . . 1
- 1.2 Research Questions . . . 2
- 1.3 Dissertation Structure . . . 2

**2 Related Work** 4
- 2.1 LLM-based Tutoring Systems . . . 4
- 2.2 From Static Prompt Optimization to Prompt Routing . . . 5
- 2.3 AI Feedback and Feature-based Reward Learning . . . 5
- 2.4 Bridging the Sim-to-Real Gap and Score Alignment . . . 6

**3 Methods** 7
- 3.1 Problem Formulation . . . 7
- 3.2 System Framework . . . 7
- 3.3 Input Representation ($X$) . . . 9
- 3.4 Prompt Space ($P$) . . . 9
- 3.5 Environment Simulation and User Modeling . . . 10
  - 3.5.1 Data-Driven Student Modeling . . . 10
  - 3.5.2 Instructional Scenario Generation . . . 11
- 3.6 Pedagogical Score ($Y$) . . . 12
  - 3.6.1 Feature Extraction via LLMs . . . 12
  - 3.6.2 Correlation-based Feature Weighting . . . 12
  - 3.6.3 Score Smoothing and Calibration . . . 14
- 3.7 Routing Model . . . 15
  - 3.7.1 Model Architecture . . . 16
  - 3.7.2 Optimization Algorithm: PPO with Experience Replay . . . 17

**4 Experiment Setup** 18
- 4.1 Hyperparameters and Training Details . . . 18
- 4.2 Simulation Benchmark . . . 18
- 4.3 Real-World Deployment and Evaluation Methodology . . . 19

**5 Experiment Results** **21**
5.1 Diversity of Synthetic Dialogue . . . 21
5.1.1 Impact of Student Profiles . . . 21
5.1.2 Impact of Tutor Strategies . . . 22
5.2 Consistency of Feedback Features . . . 24
5.3 Result of Training . . . 26
5.3.1 Overall Learning Convergence in Simulation . . . 26
5.3.2 Decoupling of Subject Representations . . . 27
5.3.3 Score Stratification by Environment and Strategy . . . 28
5.3.4 Emergence of Subject-Specific Strategies . . . 29
5.3.5 Empirical Alignment of the Learned Policy . . . 30
5.4 Simulated Benchmark: Comparative Analysis . . . 32
5.4.1 Overall Performance and Statistical Significance . . . 32
5.4.2 Robustness Across Student Profiles . . . 33
5.4.3 Behavioral Mechanisms of Success . . . 33
5.5 Real-World Deployment and A/B Testing . . . 35
5.5.1 Pedagogical Quality and Robustness . . . 35
5.5.2 Analysis of Interaction Dynamics and Efficiency . . . 36
5.5.3 Strategy Dynamics and Learning Evolution . . . 38
5.5.4 Subject-Specific Evolution and Strategy Alignment . . . 41
5.5.5 Sim-to-Real Trajectory and Bias Correction . . . 43
5.6 Beyond Score: Mapping to Educational Outcomes . . . 46
5.6.1 Correlation Between Pedagogical Behaviors and Student Success 46
5.6.2 Driving Learning Engagement . . . 47
5.6.3 Impact on Instructional Performance . . . 49
5.7 Qualitative Analysis: Real-World Scaffolding . . . 50

**6 Discussion and Conclusion** **52**
6.1 Answers to Research Questions . . . 52
6.1.1 RQ1: Score Validity . . . 52
6.1.2 RQ2: Adaptation and Sim-to-Real Transfer . . . 52
6.1.3 RQ3: Education Impact . . . 53
6.2 Summary of Research Findings . . . 53
6.3 Limitations . . . 54
6.4 Future Work . . . 55
6.5 Conclusion . . . 55

**A Simulated Conversation Transcripts** **60**

**B System Prompt Configurations** **67**
B.1 Production Baseline Prompt (Abstracted Blueprint) . . . 67
B.2 Pedagogical Prompt Space . . . 68
B.3 Literature-Based Baseline (Dinucu_Prompt) . . . 70
B.4 Linguistic Style Analysis Prompt . . . 71
B.5 Student Simulator Persona (Unmotivated Profile) . . . 71

**C Score Modeling and Evaluation Rubrics** **73**
C.1 Pedagogical Evaluator Prompt . . . 73

C.2 Pedagogical Labeling Guidelines (14 Criteria) . . . . . 73

# Chapter 1

# Introduction

In this chapter, we outline our problem statement, research questions, and the overall structure of this thesis.

## 1.1 Problem Statement

Large Language Models (LLMs) have emerged as a powerful tool in the field of education [1] [2] [3], particularly for constructing personalized tutoring systems. This research involves a commercial education platform StudyGo from the company FutureWhiz, designed for Dutch high school students. This platform covers curriculum materials across 13 different subjects, such as math, French, and history, allowing students to learn and practice various topics through interactive dialogues with an AI tutor (see Figure 1.1).

However, a challenge in this platform is the lack of a systematic mechanism to optimize instructional quality dynamically. The current approaches rely on static prompting [4], which can not adapt to different educational subjects. Furthermore, given the vast educational content, it is impractical to manually design strategies for every specific subject or topic. Moreover, an additional challenge in real-world deployment is that students often skip follow-up practice exercises if they feel the AI conversation alone has provided sufficient practice. This drop-off behavior creates a missing data problem, depriving the prompting router of the direct educational outcomes (e.g., exercise scores) needed as signals.

Moreover, most existing research validates pedagogical efficacy within synthetic environments [5, 6]. While the simulation provides safe exploration, the idealization often overlooks the distributional shift, data sparsity, and complex interaction dynamics in real-world education [7].

Therefore, solving these problems, it requires not only a reliable metric for instructional quality but also an adaptive strategy capable of generalizing across diverse contexts for real students. Specifically, we can formulate our research question as: **"Can we design an adaptive tutoring system that can improve learning outcomes across different subjects for real students?"**

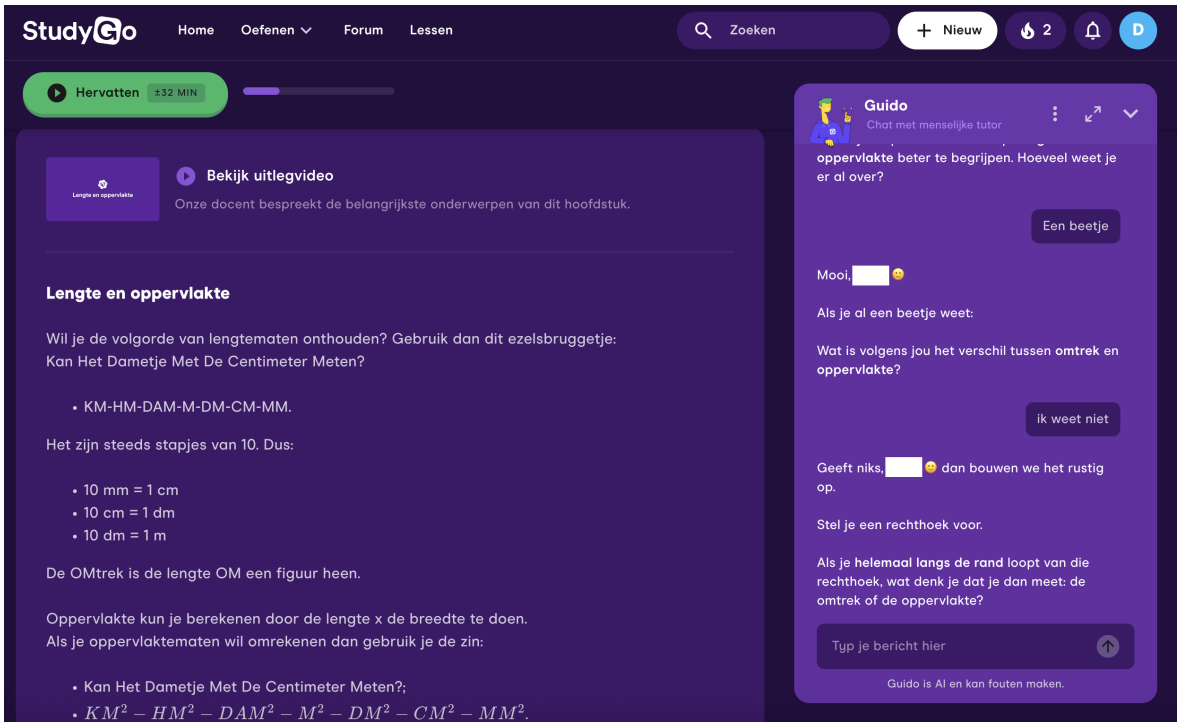



Figure 1.1: A screenshot of the tutoring system includes the learning material and AI tutor.

## 1.2 Research Questions

To overcome the limitations of static prompting and tackle the challenge described above, this research investigates the following key questions:

- **RQ1: Evaluation & Alignment** To what extent does AI-generated pedagogical feedback align with real-world instructional quality, and how do these process-based metrics map to immediate student learning outcomes?
- **RQ2: Adaptation & Sim-to-Real Transfer** Can a Subject-aware Prompt Routing framework effectively learn an adaptive teaching strategy in simulation, and how does the strategy evolve when deployed to handle sparse, real-world student interactions?
- **RQ3: Educational Impact** Does the dynamic, subject-aware prompt selection impact real-world student engagement (e.g., exercise conversion rates) and performance consistency compared to static baselines?

## 1.3 Dissertation Structure

This thesis is organized into six chapters. Chapter 1 introduces the problem statement and primary research questions. Chapter 2 reviews related work, including LLM-based tutoring systems, the transition from static prompt optimization to contextual selection, reinforcement learning from AI feedback (RLAIF), feature-based score learning and social learning, and finally, strategies for bridging the simulation-to-reality gap in

score alignment. Chapter 3 details the problem formulation and the proposed system framework integrating routing model and LLMs. It formally defines key elements, such as the feedback score mechanisms, and the routing model, and concludes with the experimental settings and implementation details.

Following the methodology, Chapter 4 presents the experimental results, including simulation outcomes, score consistency analysis, router training convergence, and benchmarking against static baselines, followed by empirical findings from the real-world A/B deployment. Chapter 5 provides a deeper discussion of these results, exploring the Sim-to-Real gap and analyzing the architectural trade-offs between macro-disciplinary and micro-topical adaptation. Finally, Chapter 6 summarizes the conclusions of this research and outlines potential directions for future work.

# Chapter 2

# Related Work

This chapter introduces the related work, including LLMs tutoring system, prompt optimization, AI feedback, feature-based score learning, and sim-to-real gap. Those topics are elements of this thesis.

## 2.1 LLM-based Tutoring Systems

The potential of Large Language Models (LLMs) to transform personalized learning has been recognized in recent literature. For instance, Maity and Deroy (2024) [8] explored how LLMs can be integrated into the educational process achieving personalization. Moreover, recent research emphasizes the importance of curriculum-specific adaptation. For instance, Malik et al. (2025)[3] demonstrated that LLMs can effectively generate high-quality curriculum scaffolds for middle school mathematics, providing structured pedagogical support to adapt materials for student needs.

Moreover, recent studies explored Reinforcement Learning (RL) to enhance the educational quality of LLM-based tutoring systems. For instance, Nam et al. (2025) [5] proposed an efficient RL framework that optimizes conversation-level outcomes (e.g., student solve rates) by mapping dialogue history into a low-dimensional latent input representation to select high-level tutoring actions. Similarly, Dinucu-Jianu et al. (2025) [6] introduced an online RL alignment method using student simulators, training the LLM to adopt a Socratic teaching style while strictly penalizing behaviors that reveal solutions directly (i.e., answer leakage).

However, while these RL approaches show promise, they usually make decisions turn-by-turn and rely on manually adjusted weights to balance different goals, like teaching quality versus task completion. Because these manual settings lack real-world evidence, the AI's goals may not align with actual human teaching values.

## 2.2 From Static Prompt Optimization to Prompt Routing

The performance of Large Language Models (LLMs) is sensitive to the system prompts. Therefore, to design effective prompts, early methods such as Automatic Prompt Engineer (APE) [9] and OPRO [10] utilized LLMs to search for a single optimal prompt. Moreover, in educational application, the prompt must be designed to facilitate learning; for instance, Abolnejadian et al. (2024) [11] demonstrated that prompt-based instruction in computer science course can enhance learning outcomes.

However, a static prompt often fails to adapt to the diverse subjects (e.g., mathematics, history). To address this, recent research has modeled prompt selection as a Contextual Bandit (CB) problem, where the optimal action is based on an observed context. A study by Hu et al. (2025) [12] introduced the PAK-UCB algorithm, emphasizing that model selection must be "prompt-aware" by using kernel-based functions to predict scores for unseen contexts.

Meanwhile, the concept of routing has emerged as a prominent strategy in LLM inference. While most existing work focuses on model routing, selecting between different LLMs to optimize the cost-performance trade-off [13, 14] or utilizing bandit-based optimization for efficient inference [15, 16]. Moreover, Singh (2026) [17] demonstrated that prompt routing can enhance task-specific accuracy, such as in arithmetic reasoning. Compared to previous work, our prompt routing is utilized to enhance learning qualities across different subjects.

## 2.3 AI Feedback and Feature-based Reward Learning

Reinforcement Learning from Human Feedback (RLHF)[18][19] has proven successful in aligning Large Language Models (LLMs) with human preference; however, the scarcity of high-quality labeled data remains a primary bottleneck. To address the high cost and scalability limitations of human labeling, the concept of Constitutional AI (CAI) was introduced by Bai et al. 2022 [20]. AI feedback can offer a scalable alternative to costly human feedback.

Recent studies by Lee et al. (2024) [21] demonstrated that RLAIF achieves performance comparable to RLHF in summarization and dialogue generation, suggesting that AI preferences can serve as an alternative to humans. Furthermore, they proposed direct RLAIF (d-RLAIF), a technique that avoids traditional reward model training by directly querying rewards from pre-trained LLMs. Their results indicate that d-RLAIF not only simplifies the alignment pipeline but also yields superior performance compared to standard RLAIF in certain zero-shot settings.

Furthermore, Schultner et al. (2025) [22] proposes Social Feature Learning (SFL), which argues that adaptive behavior emerges from learning the associations between specific observations and their respective underlying features. Our research combined the concepts above to directly generate 14 pedagogical features from LLMs, instead of a single

black-box score.

## 2.4 Bridging the Sim-to-Real Gap and Score Alignment

An important challenge in LLM-based tutoring systems is the sim-to-real gap [23], where models tend to over-optimize noisy or idealistic score signals from simulators, a phenomenon known as reward gaming [7]. As identified by Skalse et al. (2022) [24], misalignment between the simulator's score function and the actual human intent can lead to sub-optimal policies that fail in real-world deployments.

Recent work from Mao et al. (2024) [25] emphasizes that score calibration is critical for grounding simulated feedback in real-world performance. Inspired by this insight, our research addresses the sim-to-real difference through an empirical calibration of the score signal. Following robust decision-making [26], we implement a score smoothing mechanism to align the score distributions between simulated environments and real-world deployment.

# Chapter 3

# Methods

## 3.1 Problem Formulation

In this thesis, we address three core objectives: (1) quantifying the pedagogical quality of LLM-based tutoring through an LLM evaluator (addressing RQ1), (2) leveraging this measurement to implement an adaptive algorithm that optimizes instructional efficacy in simulation (addressing RQ2), and (3) ensuring that this promt routing mechanism can effectively adapt to live deployment and improve student engagement or learning outcome (addressing RQ2 & RQ3).

To formalize these objectives, we model the tutoring prompt selection task as a Contextual Bandit (CB) prompt routing problem. This framework is formally defined by the tuple $\langle X, P, Y \rangle$, where the router observes a topic context and subject ID $x \in X$, selects a tutoring strategy $p \in P$, and receives a feedback score $y \in Y$ based on the interaction quality. Therefore, this framework can learn the optimal mapping between observed pedagogical context and tutoring strategy $\pi : X \to P$ that maximizes the expected cumulative pedagogical score $Y$.

## 3.2 System Framework

Figure 3.1 illustrates the overall architecture of our Subject-aware Prompt Routing system. The framework is designed to implement the Contextual Bandit formulation defined in Section 3.1. The workflow consists of three primary phases: (1) **Input Encoding and Prompt Selection**, where the topic context and subject ID ($x$) are mapped to a semantic space; (2) **Dialogue Interaction**, where the student interacts with the LLM-tutor guided by the selected prompt $p$; and (3) **AI Feedback Generation**, where the LLM evaluator provides a pedagogical signal ($y$) based on the conversation transcript. This framework provide a closed learning loop to optimize the routing model, even when an explicit feedback signal (e.g. exercise score) is missing.

**Input Encoding and Prompt Selection** As illustrated in Figure 3.1, the process begins with the input representation ($X$), where the system extracts the Topic Title and the Subject ID for each tutoring session. The routing model computes a probability

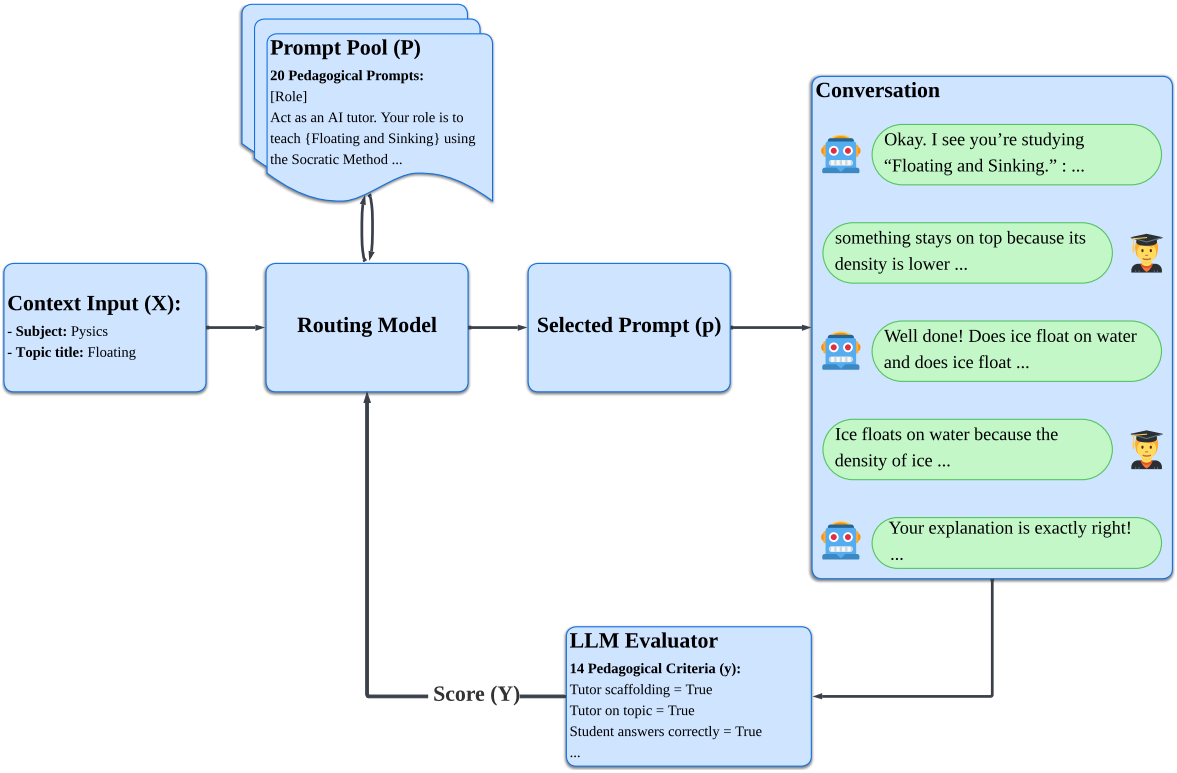



Figure 3.1: System Framework of Subject-aware Prompt Routing. The tutoring system encodes subject-specific semantic input ($X$) to select pedagogical strategies ($p$). After a multi-turn dialogue, an LLM-based evaluator decomposes the interaction into 14 pedagogical criteria ($y$) to compute a feedback signal ($Y$) for prompt optimization, based on the conversation transcript.

distribution over the Prompt Space ($P$). A prompt $p$, representing a specific instructional prompt selected from the pool of 20 strategies, is sampled from this distribution.

**Dialogue Interaction** The selected prompt is fed into the LLM (GPT-5-mini) to serve as an AI tutor interacting with the student. For simulation, we design LLM-based students to interact with an AI tutor, generating a multi-turn dialogue transcript. In the live deployment, the AI tutor will directly instruct real high school students.

**AI Feedback Generation** Finally, the LLM evaluator analyzes the transcript to assess instructional quality. It provides feedback features $c \in C$ using a feature-based mechanism, evaluating 14 granular pedagogical dimensions. The final feedback signal ($Y$) is calculated by the calibration function and used to update the router, effectively closing the learning loop. By decoupling the prompt pool ($P$) from the routing model and conditioning selection on the semantic input ($X$), the framework enables dynamic strategy selection that adapts to the specific subject.

**Example** To illustrate, when the student asks about the topic "Product, quotient, sum, and difference" in math, the router receives the encoded topic title and subject ID to choose a prompt from the prompt pool. After interaction, an LLM evaluator

will generate educational features, and then we formulate those features into a final pedagogical feedback score to optimize the router.

## 3.3 Input Representation ($X$)

The input representation ($x \in X$) consists of topic title and subject ID, enabling the routing model to observe the pedagogical scenario.

To encode the context of topic title, we utilized a frozen pre-trained encoder (E5-base-V2) $E_\phi$ to map raw context into 768-dimensional vector space. However, in our preliminary experiment, we observed the representation collapse when directly using a pre-trained encoder. As illustrated in Figure 3.2, the cosine similarity between the centroids of different subjects is consistently high ($> 0.92$), with many pairs exceeding 0.96. This indicates a severe anisotropy or representation degeneration problem [27, 28] , where semantic vectors are clustered within a narrow cone. Such a collapsed semantic space [29] makes it difficult for the router to distinguish between distinct topic context, potentially hindering the router's ability to specialize its instructional strategies.

To overcome this limitation, we introduced a trainable Subject Embedding layer $E_\psi$. This layer maps discrete subject ID into a 64-dimensional continuous vector space, acting as learned anchors that represent the unique pedagogical requirements of each discipline. The final input representation $x$ is formed by concatenating these components and applying Layer Normalization to stabilize the input distribution:

$$x = \text{LayerNorm}(E_\phi(\text{topic_title}) \oplus E_\psi(\text{subject_id})) \tag{3.1}$$

where $\oplus$ denotes vector concatenation. This encoding strategy ensures that the router receives both lower-level topic context with subject ID, unfolding the collapsed latent space for subject-specific adaptation.

## 3.4 Prompt Space ($P$)

In our framework, the prompt space comprises 20 pedagogical prompts with different instructional strategies (e.g., Socratic scaffolding, Direct explanation). The foundational prompts were explicitly authored by human educational experts from FutureWhiz. We utilized LLMs to increase variation (App. B.2) based on expert-crafted baseline prompt (App. B.1). Moreover, to prevent the unsafe instructional behavior (e.g., off-topic) of AI in the production environment, all 20 prompts share the same behavior constraints.

The baseline prompt template consists of role, interface capability, pedagogical instructions, and behaviors. The instructional style of baseline is the Socratic method, which uses cooperative, question-based dialogue to help students discover the answer themselves rather than directly providing answers. The other iconic instruction method is Feynman in the prompt pool. This method emphasizes the concept of role reversal in the teaching process, where students act as instructors to explain problems in their own words, identifying knowledge gaps and solidifying their understanding.

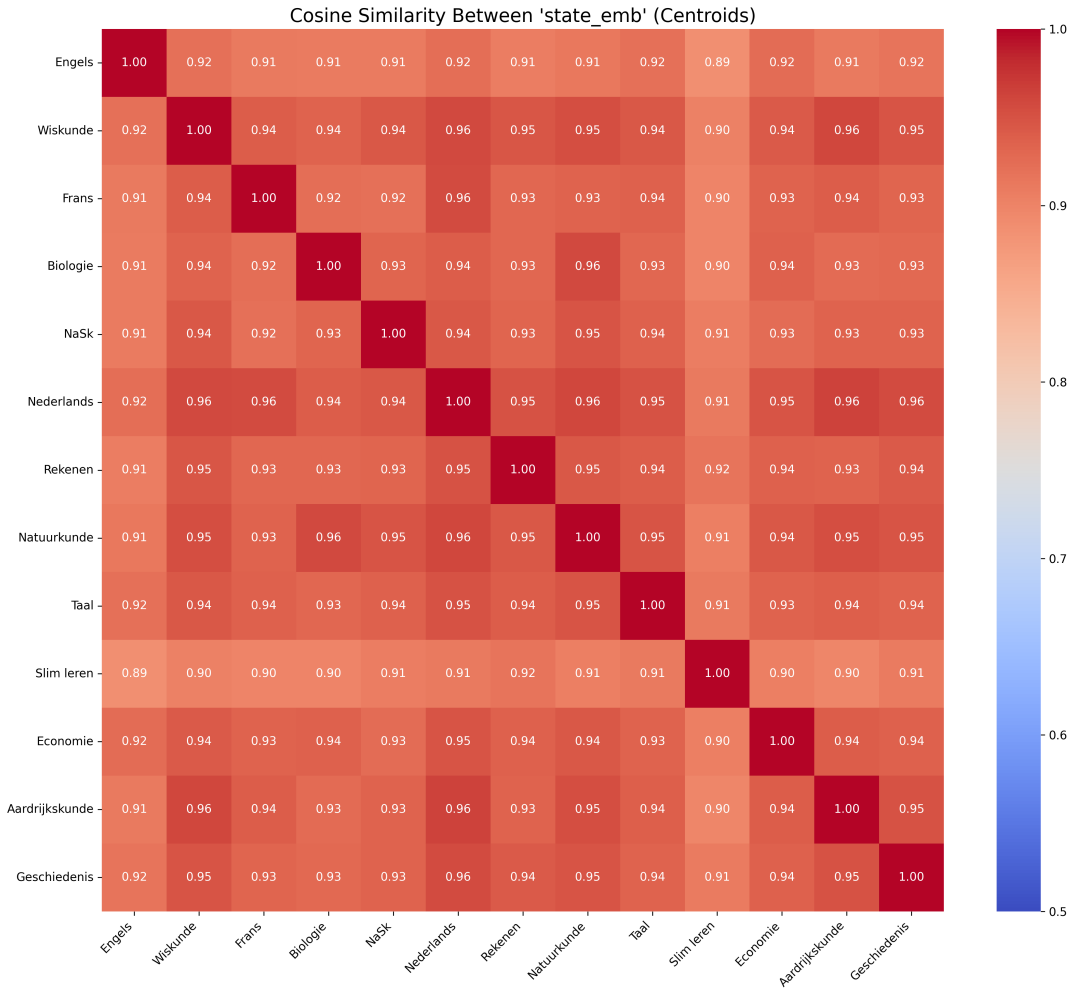



Figure 3.2: Cosine Similarity Heatmap of Title-based Input Embeddings. The visualization reveals a highly collapsed latent space with average similarities exceeding 0.92 across disparate subjects. This empirical evidence of semantic anisotropy justifies our dual-path architecture, which utilizes a trainable subject embedding layer to restore the discriminative power required for subject-specific prompt selection.

## 3.5 Environment Simulation and User Modeling

To facilitate the effective training of the routing model without the risks associated with exposing real students to an untested router, we developed a high-fidelity simulation environment (Conversation box in Figure 3.1) driven by Large Language Models (LLMs). This environment comprises two interconnected subsystems: a **Data-Driven Student Simulator** and an **Instructional Scenario Generator**, which collectively provide a robust training ground for model optimization.

### 3.5.1 Data-Driven Student Modeling

Unlike real user, simulators that often exhibit robotic or formal behavior, our student routers are grounded in real-world educational data to ensure behavioral realism. We categorize student behaviors into three distinct profiles: *Motivated*, *Mediocre*, and *Un-*

*motivated*, derived from a qualitative analysis of anonymized chat logs ($\mathcal{D}_{chat}$) collected from the commercial education platform FutureWhiz (See the chat box in Figure 1.1). The detailed system prompt templates used to define these personas are provided in Appendix B.5.

To capture the linguistic nuances specific to Dutch high school students, such as slang, informal tone, and variable sentence structure. We employ a few-shot prompting strategy for style transfer. At the initiation of each simulation session, the system samples $N = 3$ real conversation transcript. A *Linguistic Analyst*, using LLM GPT-5-mini, then extracts a "style summary" from these samples. This summary captures the specific persona's communicative patterns and is used to condition the Student Simulator, ensuring it mimics real adolescent responses rather than standardized model outputs. The specialized system prompt used to guide this style extraction process is detailed in Appendix B.4.

Furthermore, each student's profile is governed by specific system prompts, defining LLM student behavior; for instance, the *Unmotivated* student is explicitly prompted to exhibit low patience, easily to distraction, and a tendency to deviate from the topic, thereby challenging the Routing Model to maintain pedagogical engagement under adversarial conditions.

### 3.5.2 Instructional Scenario Generation

To ensure the router learns to generalize across diverse educational subjects, we implemented a procedural generation pipeline for learning contexts. For each training episode, a topic is sampled from a comprehensive curriculum database ($\mathcal{D}_{topic}$), providing the router with the subject matter and topic title.

The dynamics of the interaction are stochastically determined based on the selected student profile. *Motivated* students are configured for extended engagement (3–15 turns), whereas *Unmotivated* students simulate realistic attrition with shorter interactions (1–5 turns). To further refine the simulator's output toward realistic brevity and imperfection, we inject "inner monologues" into the prompt stream (e.g., `[Think: you are confused; reply in one short line]`). This technique, inspired by the reasoning traces in Chain-of-Thought prompting [30], acts as a soft constraint that prevents the simulator from generating the unrealistically verbose or perfectly structured responses typical of base LLMs. The mechanism for this dynamic constraint injection and examples of latent thoughts are further detailed in Appendix B.5.

During each simulated session, the Routing Model selects a prompt $p \in P$ corresponding to a specific pedagogical strategy. To execute this strategy, the environment employs a parameterized tutor prompt that integrates the sampled curriculum variables alongside the real-time conversation history. These prompts impose strict behavioral constraints, forcing the LLM to adopt the selected pedagogical persona, such as Socratic scaffolding or direct analytical breakdowns. While the specific instructional constraints within the prompt pool are part of a proprietary commercial framework, the architecture ensures that each prompt $p$ consistently translates into a distinct and measurable instructional behavior. A structural abstraction of these instructional prompts, illustrating the core pedagogical logic and behaviors, is presented in Appendix B.1.

The resulting dataset $\mathcal{D}_{sim}$ consists of $(x, p, y)$ tuples covering a wide distribution of student behaviors and academic topics. By generating comprehensive session-level trajectories, the simulation environment provides a stable foundation for the prompt routing framework.

## 3.6 Pedagogical Score ($Y$)

To align the router's objective with human pedagogical values, we construct the pedagogical score, inspired by Reinforcement Learning from AI Feedback (RLAIF) [21] and Social Feature Learning (SFL) [22]. The feedback score ($y \in Y$) generation process consists of two stages. First, we employ a Large Language Model (GPT-5-mini) as an automated evaluator to analyze the conversation transcript and perform multi-dimensional binary classification. The evaluator detects the presence or absence of predefined pedagogical features (e.g., understanding, curiosity), which are then aggregated into a weighted raw score. Second, we apply a non-linear function to smooth this raw score. This smoothing mechanism serves a dual purpose: it calibrates the empirical score distribution to more closely align with real user feedback patterns, while simultaneously stabilizing the learning signal to mitigate gradient variance during model optimization.

### 3.6.1 Feature Extraction via LLMs

To guarantee robust and consistent feature extraction, we designed a structured evaluation prompt. This prompt supplies the LLM evaluator with the complete interaction context, encompassing the academic subject, topic title, source educational material, and the full chat transcript. Furthermore, the prompt explicitly defines the evaluator's persona, imposes strict output constraints, and describes the evaluation guidelines.

In total, we formulated 14 binary criteria (see Section 3.6.2) to evaluate the student's learning state and the tutor's instruction. To mitigate subjective variance, we defined each criterion by providing explicit rationales and conditional triggers. For instance, to evaluate the criterion of 'effective scaffolding', the prompt explicitly defines the classification boundaries: it evaluates to True if the tutor offers progressive, multi-turn guidance that helps the student reason step-by-step and adjusts the level of help when the student struggles. Conversely, it is strictly evaluated as False if the tutor gives full answers immediately, only asks questions without supporting reasoning, or provides a single explanation without adapting over multiple turns. This rigorous specification is applied across all 14 criteria. To ensure consistent feature extraction, we developed a structured evaluation prompt (see Appendix C.1) that guides the LLM. The specific binary classification boundaries and pedagogical rationales for all 14 criteria are detailed in Appendix C.2.

### 3.6.2 Correlation-based Feature Weighting

Instead of assigning arbitrary scalar weights to the evaluation criteria, we derive the importance of each feature empirically. This approach is motivated by the challenge of accurately modeling human preferences in score functions, as highlighted by Knox

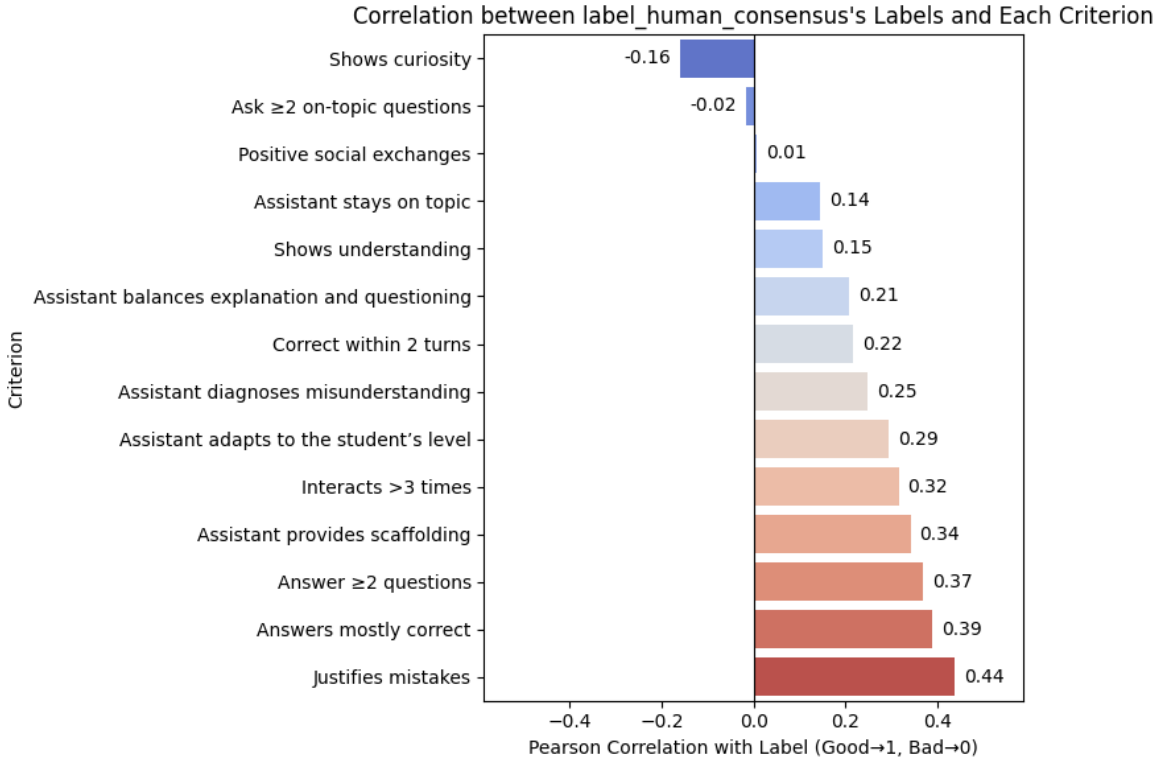



Figure 3.3: Pedagogical Feature Correlation from Expert's Consensus. The result shows that correctness has a higher correlation to pedagogical success. However, while showing curiosity is seen as a positive behavior in education, it may indicate a student asking off-topic questions in the conversation.

et al. (2022) [31], who emphasize the importance of aligning score models with actual human evaluative intent.

In our research, the weights $w_i$ are derived from Pearson correlations with human-labeled learning outcomes. We correlated the 14 LLM-extracted features against an expert-labeled dataset ($N = 138$). Human experts evaluated sessions and label it to 0, 0.5, and 1. These coefficients serve as the weights $w_i$ for aggregating the raw score $y_{raw}$.

As shown in Table 3.3, features such as Justifying mistakes ($w = 0.436$) and Answers mostly correct ($w = 0.387$) show the strongest positive correlation with learning quality, whereas Showing curiosity exhibits a slight negative correlation ($w = -0.16$), possibly indicating student confusion in this specific context. This result suggests that correctness is more important to reflect educational success. While showing curiosity is a positive behavior in education, it may indicate that the students are asking off-topic questions. In our implementation, the negative correlation is set to 0 for weights, avoiding the uncertainty of the positive behavior.

The raw score $y_{raw}$ is calculated as the weighted sum of these binary features $c_i \in \{0, 1\}$:

$$y_{raw} = \sum_{i}^{14} w_i \cdot c_i \tag{3.2}$$

where $y_i \in \{0, 1\}$ represents the binary presence of the $i$-th feature as detected by the LLM evaluator, and $w_i$ denotes the corresponding Pearson correlation coefficient derived from human-labeled pedagogical benchmarks.

### 3.6.3 Score Smoothing and Calibration

Raw pedagogical scores by linear sums can be noisy. To normalize the score distribution and control the optimization landscape, we apply a sigmoid-based smoothing transformation to $y_{raw}$:

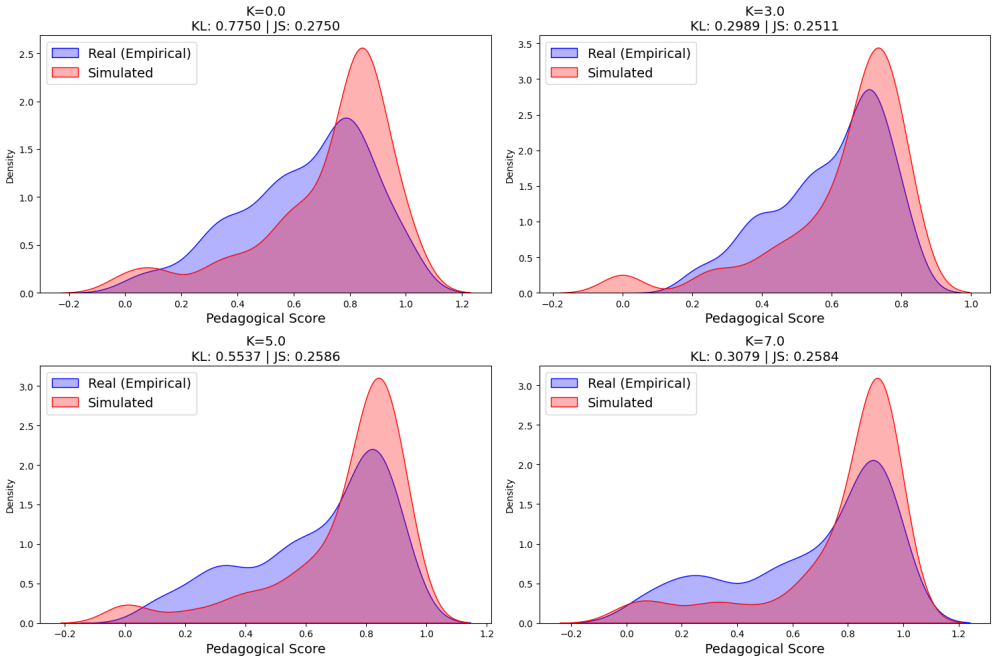
$$y_{smoothed} = \frac{1}{1 + \exp(-K \cdot (y_{raw} - 0.5))} \tag{3.3}$$

where $K$ is a smoothing coefficient controlling the sensitivity of the feedback signal.

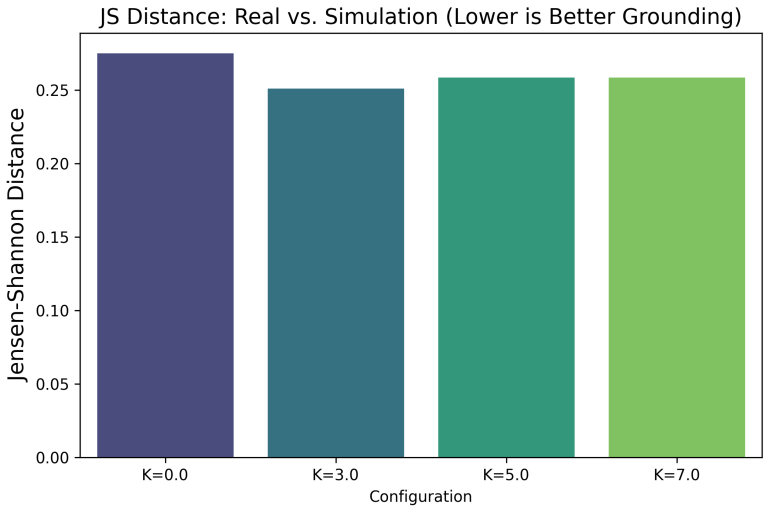
To bridge the gap between the real world and simulation, we used a smooth function and calibrated the smoothing parameter $K$. By comparing the distribution of the real-data and simulation, we found that $K = 3.0$ can minimize the Jensen-Shannon (JS) distance, compared to $K = 0.0$ (Figure 3.4).

As illustrated in Figure 3.4b, we compared $K \in \{0, 3, 5, 7\}$, and $K = 3.0$ achieved the lowest Jensen-Shannon (JS) distance, indicating the best alignment between simulated scores and empirical human labels. Furthermore, Figure 3.4a displays the score density distributions. While $K = 0.0$ results in a distributional shift, $K = 3.0$ grounds the simulation by mimicking the empirical score density observed in real tutoring sessions.

The selection of $K = 3.0$ alleviates the distributional shift [7] between simulation and real-world. This inconsistency introduces the model over-optimization for the simulation environment [32, 33]. Consequently, we fixed $K = 3.0$ for all experiments.




(a) score Density Comparison across different $K$ values




(b) Quantitative alignment error (JS Distance)

Figure 3.4: Empirical Score Calibration for Bridging the Sim-to-Real Gap. (a) Visualization of simulated score densities compared to human ground-truth; (b) JS Distance results showing that $K = 3.0$ achieves the minimum distributional shift, providing stable model learning and avoiding model overfitting for the simulation.

## 3.7 Routing Model

Given the input representation $x$ and the discrete prompt space $P$, the routing model acts as the decision-making agent. To map pedagogical contexts to optimal instructional strategies, we design a custom neural network architecture. This section details both the network design and the optimization process.

### 3.7.1 Model Architecture

As illustrated in Figure 3.5, our routing model employ an Actor and Critic architecture designed to maintain subject-specific sensitivity. The model utilizes frozen topic embeddings and trainable subject embeddings to observe topic and subject information, followed by decoupled MLP heads for prompt selection and value estimation.

A defining feature of this architecture is the Residual Subject Injection mechanism. As shown by the skip-connections in Figure 3.5, the subject embedding $v_{subj}$ is not only provided as an initial input but is also directly injected into the final Actor head and Critic head. This design choice, implemented in our Actor-Critic routing model, ensures that categorical subject information remains a primary determinant of the output probability distribution. By bypassing the primary MLP layers, this injection prevents the specific pedagogical requirements of the subject from being diluted or lost during the deep non-linear feature extraction process, which is particularly critical given the high similarity of topic titles identified in Figure 3.2.

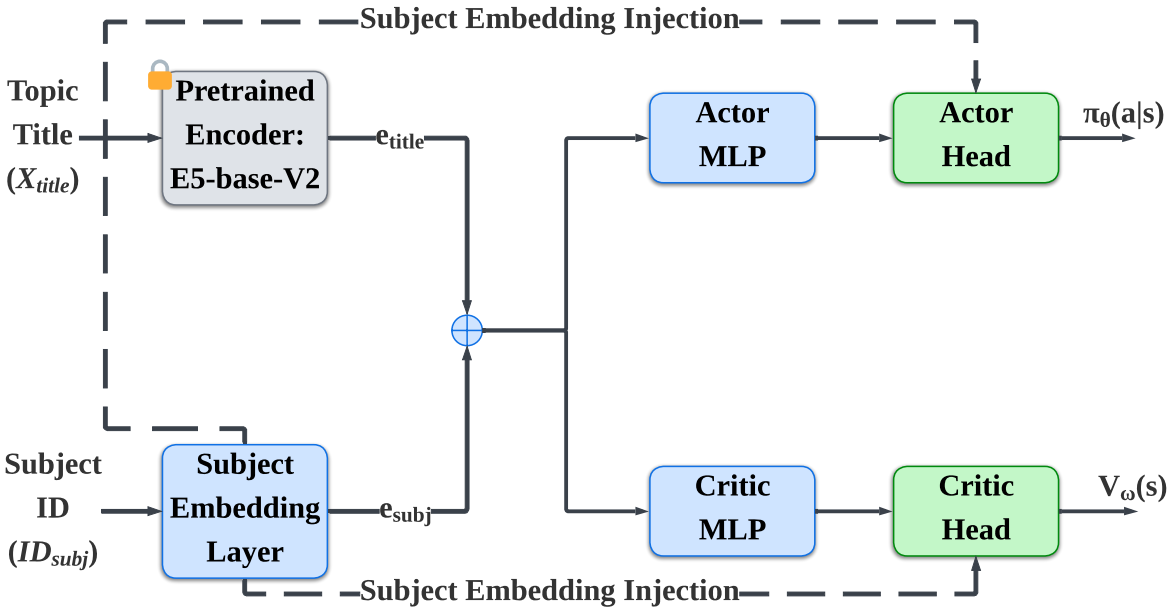



Figure 3.5: Detailed Architecture of the Actor-Critic Routing Model. The framework integrates frozen semantic features with trainable subject embeddings via a Residual Subject Injection mechanism. By injecting subject-specific anchors directly into the final strategy and value heads, the model ensures robust strategy differentiation even when underlying topic titles exhibit high similarity.

The output of actor head $\pi_\theta(p|x)$ and value function $V_\omega(x)$ are parameterized by separate Multi-Layer Perceptrons (MLP) that process the normalized input vector $x$. While the actor head outputs the probability distribution over the prompt space, the value head provides a baseline estimating the pedagogical score ($y$) of the given input representation ($x$). This baseline reduces the variance during the optimization and also considers the difficulty between different subjects. Following the dual-path design, each network consists of two hidden layers with Tanh activation functions to extract nonlinear features. The dimensions of hidden layers are 512 and 256 to compress the 832-dimensional input representation (768 from topic embeddings and 64 from subject

embeddings).

To implement the Residual Subject Injection mechanism, the subject embedding $e_{subj}$ is concatenated with the extracted features $h_2$ before the final output layer. For the Actor, the forward propagation is defined as:

$$h_1 = \tanh(W_1 x + b_1), \qquad h_1 \in \mathbb{R}^{512} \tag{3.4}$$

$$h_2 = \tanh(W_2 h_1 + b_2), \qquad h_2 \in \mathbb{R}^{256} \tag{3.5}$$

$$\pi(\cdot|x) = \text{Softmax}(W_{out}[h_2 \oplus e_{subj}] + b_{out}) \tag{3.6}$$

where $W_{out} \in \mathbb{R}^{|A| \times (256+64)}$ accounts for the injected subject features, and $x$ is the input representation from the Equation 3.1 in Section 3.3.

### 3.7.2 Optimization Algorithm: PPO with Experience Replay

We formulate the optimization problem as a Contextual Bandit problem. To ensure training stability and prevent varying updates from destabilizing the decision-making, we employ Proximal Policy Optimization (PPO) [34], a policy gradient method that optimizes a surrogate objective function. While PPO is traditionally utilized for multi-step RL, restricting the episode length to a single step adapts it into a effective contextual bandit algorithm, where the actor-critic architecture naturally provides baseline-subtracted policy gradient updates.

Given that querying the LLM-based Tutor environment is computationally expensive and latency-constrained, standard on-policy PPO (which discards data after a single update) is sample-inefficient. To address this, we adopt an Online PPO framework augmented with Experience Replay. We maintain a replay buffer $\mathcal{D}$ containing transition tuples $(x, p, y, \pi_{old})$ collected from recent interactions. The objective function is defined as:

$$L^{CLIP}(\theta) = \hat{\mathbb{E}}_{(x,p,y)\sim\mathcal{D}} \left[ \min(r(\theta)\hat{A}, \text{clip}(r(\theta), 1-\epsilon, 1+\epsilon)\hat{A}) \right] \tag{3.7}$$

where $r(\theta) = \frac{\pi_\theta(p|x)}{\pi_{\theta_{old}}(p|x)}$ is the probability ratio, and $\hat{A}$ is the estimated advantage $\hat{A} = y - V_\omega(x)$.

This Actor-Critic formulation is particularly advantageous in our educational context. By utilizing the value function $V_\omega(x)$ as a input-dependent baseline, the advantage $\hat{A}$ evaluates the *relative* improvement of a prompt rather than its absolute score. This effectively accounts for the inherent difficulty of different subjects, preventing the router from being biased against naturally challenging topics.

To reuse past experiences without violating the on-policy assumption of PPO, we employ importance sampling corrections via the probability ratio $r(\theta)$. This allows us to perform multiple optimization epochs on a mixture of fresh and recent historical data, improving sample efficiency while maintaining the stability benefits of the trust region.

# Chapter 4

# Experiment Setup

## 4.1 Hyperparameters and Training Details

Based on preliminary experiments, we identified a robust set of hyperparameters that balance exploration and exploitation. We use the AdamW optimizer [35] for weight updates. For simulation benchmarking, the routing model was trained in simulation for 16,000 steps. This extended training allows us to observe the router's evolution and convergence in a controlled environment.

Moreover, for live deployment, we restricted the initial simulation pre-training to 2,000 steps and finetuned the router as learner data became available from the Real-World A/B test. This strategy provides a warm start and prevents the router from overfitting on the synthetic data. The training data is collected daily from real-user interaction and updates the routing model. Moreover, to prevent the routing model from developing a bias toward high-traffic subjects, we use a balanced sample across both the simulation and live deployment data.

A critical component of our setup is the Entropy Coefficient, which encourages the router to explore diverse prompts during the early stages of training to prevent premature convergence to a sub-optimal strategy. Additionally, we utilize Experience Replay with a 1:1 ratio of new-to-old data to improve sample efficiency. The final hyperparameters used for the main results are summarized in Table 4.1.

To encourage exploration at the start of training, the weights of the final output layer ($W_{out}$) are initialized with a low standard deviation (0.01). This results in a near-uniform initial probability distribution over actions, preventing the router from prematurely converging to suboptimal deterministic strategies. During training, prompts are sampled stochastically $p \sim \pi(\cdot|x)$, while for evaluation, the deterministic mode $p = \arg\max_p \pi(p|x)$ is used.

## 4.2 Simulation Benchmark

To evaluate the pedagogical efficacy of our dynamic approach, we benchmarked the learned routing model against two strong static baselines in the simulation environment:

Table 4.1: Hyperparameters and Implementation Details

| Parameter | Value |
|---|---|
| **Model Architecture** | |
| Encoder Model | E5-base-v2 |
| Hidden Layer Sizes | 512, 256 |
| Activation Function | Tanh |
| **Optimization (PPO)** | |
| Learning Rate | $2 \times 10^{-5}$ |
| Batch Size | 64 |
| Entropy Coefficient | 0.15 |
| Clip Range ($\epsilon$) | 0.1 |
| Gamma ($\gamma$) | 0.0 (Contextual Bandit) |
| Optimizer | AdamW |
| Experience Replay Ratio | 1:1 |

the default production prompt (Appendix B) utilized by FutureWhiz, and a literature prompt based on Dinucu et al. [6]. For this evaluation, the routing model was trained for approximately 16,000 steps, after which the network weights were frozen to ensure a deterministic decision-making during the benchmark phase.

To guarantee a fair comparison, all three prompting strategies were evaluated under identical environmental conditions. The test suite was constructed by uniformly sampling 10 topics and 3 distinct student profiles for each academic subject. This setup ensures that the metrics are not biased toward any specific discipline or learner archetype.

When interpreting these comparative results, it is essential to acknowledge a specific methodological boundary regarding the Dinucu_Prompt baseline. In the original study, the authors proposed a message-based adaptive framework. However, to maintain a strictly controlled experimental environment and isolate the effects of session-level adaptation, we only use their system prompt for their core pedagogical principles. Moreover, we introduce the variable from the platform to Dinucu_Prompt, such as subject and topic content. (See Appendix B.3)

Consequently, this benchmark does not intend to claim superiority over the entirety of Dinucu's dynamic message-level architecture in its full complexity. Instead, it serves as a focused on our RQ2: a routing model capable of dynamically selecting from a diverse prompt pool based on contextual input variables can outperform even a highly-optimized static prompt.

## 4.3 Real-World Deployment and Evaluation Methodology

For the real-world deployment, the routing model pretrained around 2,000 steps to mitigate overfitting on synthetic conversational data. To maintain an effective bal-

ance between stability and data diversity, we implemented an $\epsilon$-greedy-like sampling strategy: 20% of the prompts were sampled based on prompt probabilities to encourage exploration, while the remaining 80% utilized an greedy (argmax) selection for exploitation in the production environment.

Regarding hyperparameter configurations from Section 4.1, we maintained consistency with the simulation environment, with the exception of the batch size and the ratio of Replay Buffer. In the real-world training, the batch size was adjusted to 16 to account for data sparsity. To ensure the quality of training samples, we only included sessions with at least two conversational turns (turns $\geq 2$), as single-turn interactions lack the necessary student feedback for model optimization. To ensure robust updates, the daily training process was deferred if the volume of new daily interactions fell below a threshold of four samples. Furthermore, we employed a dynamic replay buffer strategy to manage sample distribution: if the new data volume was below the batch size but above the minimum threshold (below amounts of 4), historical data was sampled to fill the batch. When new data exceeded the batch size, an equivalent amount of historical data was included to maintain an experience ratio between 1:1 and 1:3. However, if the new data is not enough, the training process is postponed to the next day.

For performance evaluation, we conducted an A/B test in the production environment, where users were randomly assigned to either the control group (static prompt from FutureWhiz) or the variant group (router-driven dynamic prompts) in a 50/50 ratio. The primary evaluative metric was the pedagogical score. We filtered for sessions with $turns > 1$, resulting in $N = 656$ valid interactions with 359 students. Our experiments ran in March and April 2026.

Beyond the score of conversation, we analyzed the link between instructional quality in conversation and education impact (RQ3). Conversation transcripts were mapped to subsequent exercise results, using a strict temporal filter: only exercises completed on the same day and following a tutor interaction were included. We evaluated the conversion rate (the probability of a student transitioning from chat to practice) and used the Mann-Whitney U test to investigate the statistical significance between AI-generated scores and subsequent exercise accuracy or conversion rate. Furthermore, to analyze exercise accuracy, a within-user comparison was conducted for returning students to control for individual differences in prior knowledge.

# Chapter 5

# Experiment Results

## 5.1 Diversity of Synthetic Dialogue

To ensure the robustness of the router, it is crucial that the training environment provides a diverse range of instructional scenarios. We analyzed the score distributions across different synthetic student profiles and tutor strategies to verify this diversity. Qualitative examples of these simulated interactions are provided in Appendix A.

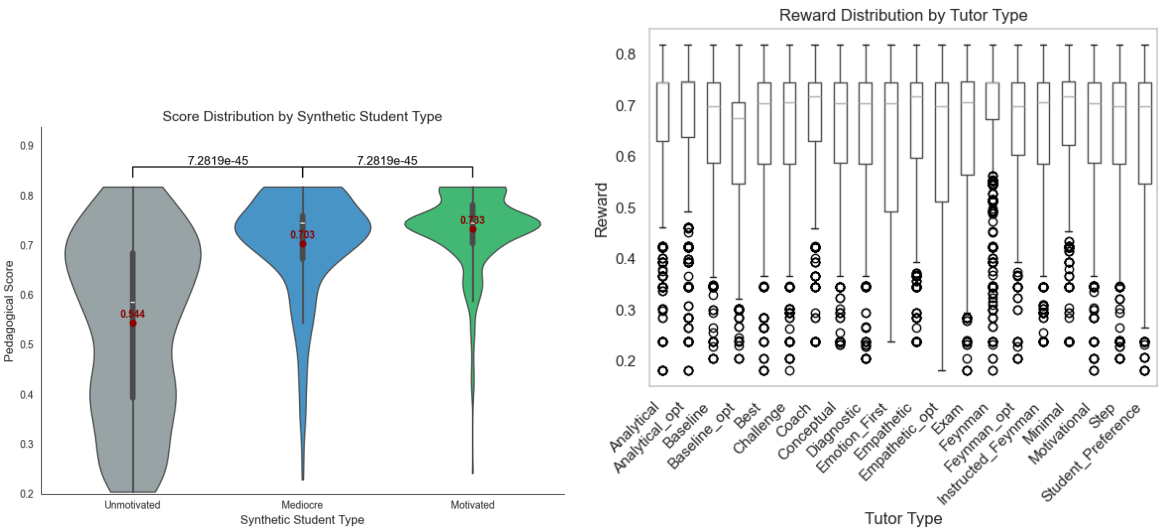
### 5.1.1 Impact of Student Profiles

To validate AI feedback for training the pedagogical features, we evaluate the interaction's pedagogical quality across three student profiles in simulation: Motivated, Mediocre, and Unmotivated. As illustrated in Figure 5.1a, the score distribution across all benchmarking methods exhibits a clear difference. High-motivation students yield the highest mean score (0.733).

Conversely, the Unmotivated student profile is characterized by a substantially lower median score (0.544) with a wider distribution, indicating that these scenarios present a consistently high barrier for the router. This contrast is important for the learning process: by exposing the model to this difficulty gradient, we ensure the routing model does not only overfit to idealized interactions but also develops the robustness required to handle uncooperative, distracted, or struggling learners.

To further validate the behavior of our LLM-based student simulator, we analyzed the trigger rates of specific evaluation criteria across different student profiles. As illustrated in Figure 5.2, the simulation effectively simulated the behavior of different levels of learners. For instance, Motivated students demonstrate a significantly higher proportion to ask on-topic questions and show understanding compared to Unmotivated students, who rarely trigger these positive engagement metrics. On the other hand, the criteria evaluating the tutor's behavior (e.g., Assistant provides scaffolding) remain consistently high across all simulated student profiles. This indicates that while the motivation of the simulated student dictates the upper bound of the pedagogical score, the tutor consistently attempts to provide structured pedagogical support.

Moreover, statistical significance was confirmed via independent t-tests, showing highly




(a) Score Distribution by Student Type (b) Score Distribution by Tutor Strategy

Figure 5.1: Analysis of dialogue diversity. (a) Score Distribution by Simulated-Student Profile. (b) Score Distribution by Different Strategies. The clear separation between "Motivated", "Mediocre", and "Unmotivated" simulated-student profiles establishes a difficulty gradient for the routing model and confirms the LLM evaluator's discriminative validity. For different strategies, while most strategies have a similar median, a prompt with stable performance can still be distinguished (e.g., *Feynman*).

significant differences between all groups ($p < 0.001$). These results confirm that the LLM evaluator can successfully discriminates between simulated-student states.

### 5.1.2 Impact of Tutor Strategies

Figure 5.1b presents the score distribution across the 20 predefined instructional prompts that constitute the router's prompt space. Compared to the variance observed among student profiles, the score distribution across tutor strategies is notably more uniform. The median score for almost every tutor type clusters around 0.7, underscoring the inherent difficulty the routing model faces in distinguishing and selecting the optimal action globally. However, slight performance variations among the strategies still exist. For instance, the *Feynman* strategy exhibits a slightly higher median with a smaller standard deviation, indicating highly consistent pedagogical quality.

While the overall score distributions appear uniform, analyzing the underlying pedagogical behaviors reveals differences among the prompts. Figure 5.3 highlights the behavioral trigger rates for the top-performing strategies alongside a baseline. It empirically confirms that our crafted prompts elicit distinct pedagogical patterns. For example, the *Analytical_opt* and *Feynman* strategies effectively guide students toward higher response correctness and understanding.

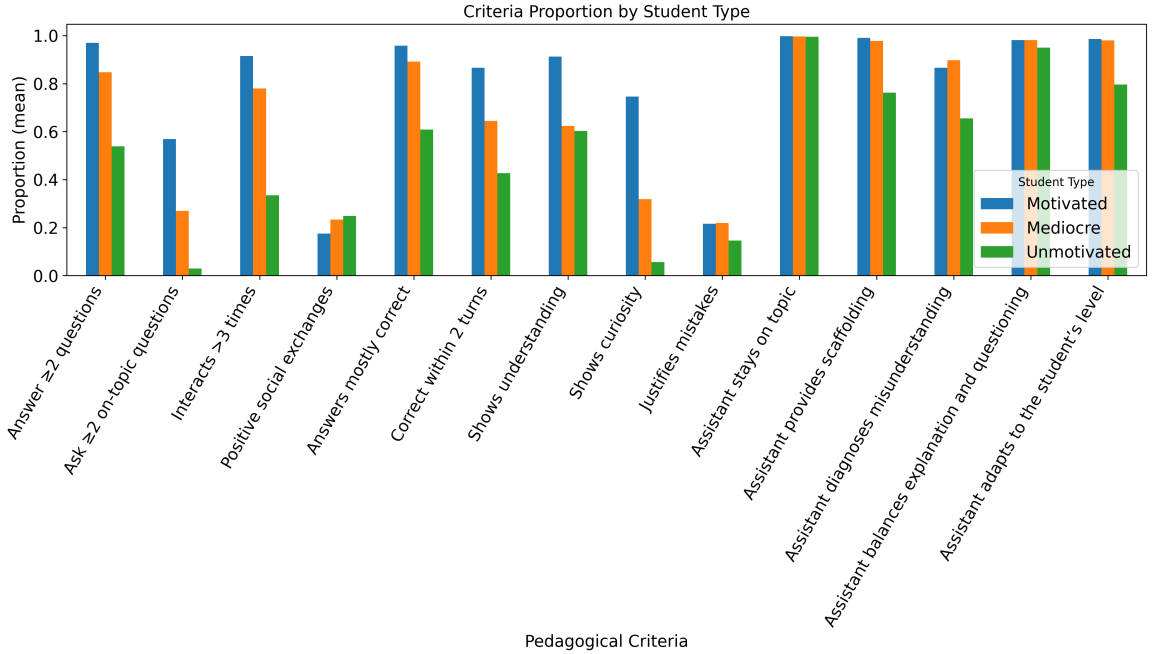



Figure 5.2: Criteria trigger rates stratified by simulated student profile. The behavioral divergence (e.g., in asking questions and showing understanding) confirms the diversity of the LLM-based student simulator, while consistent tutor scaffolding highlights the robustness of the baseline teaching interactions.

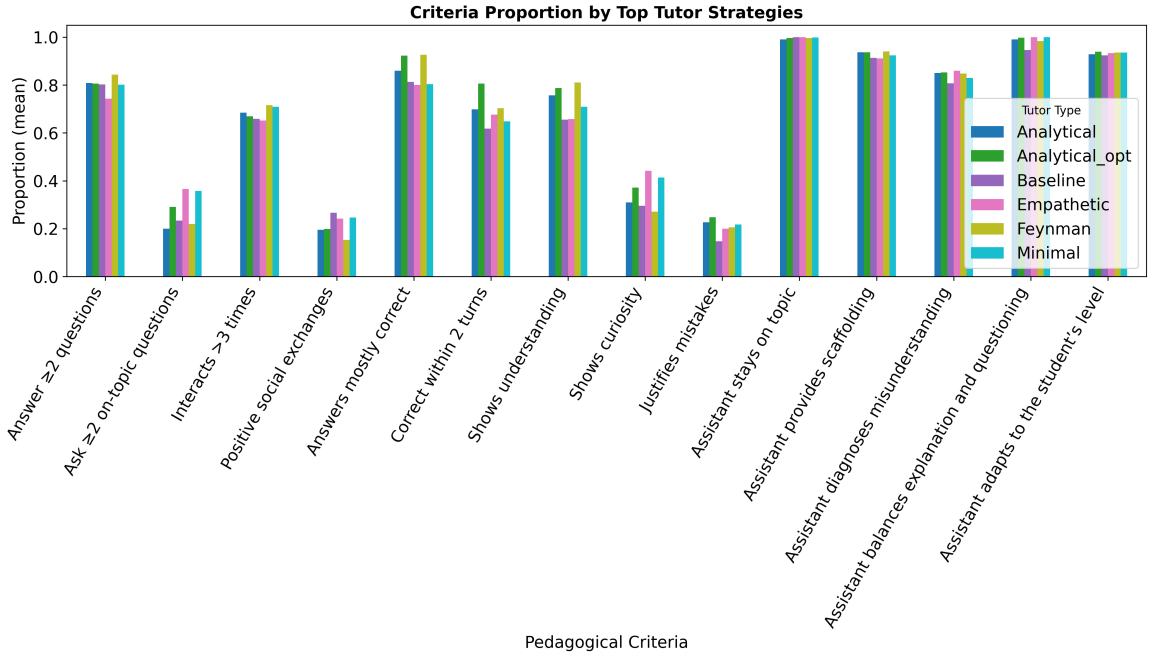



Figure 5.3: Criteria trigger rates of the top 5 tutor strategies. Despite similar overall score medians, the distinct variations in specific criteria triggers (e.g., answers mostly correct and shows understanding) prove that the prompts introduce fundamentally different teaching behaviors.

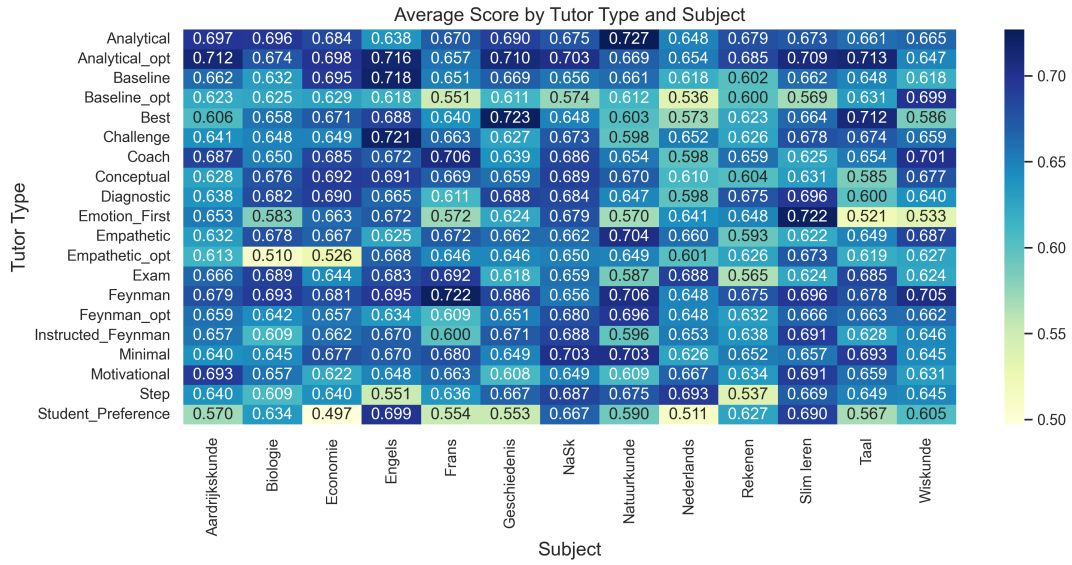
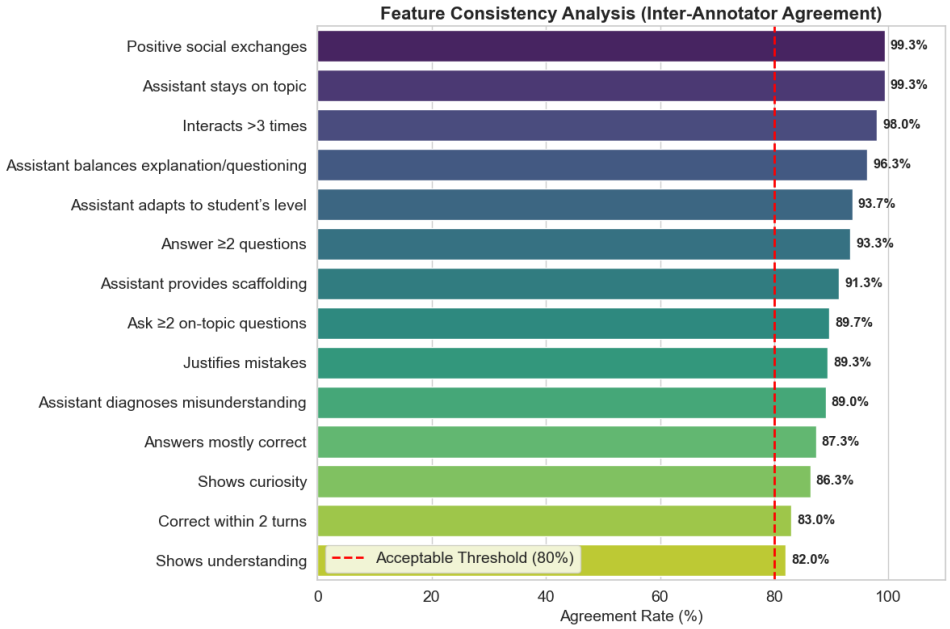



Figure 5.4: Heatmap of average session scores categorized by tutor strategy and academic subject. Darker blue cells indicate higher average scores. The varied effectiveness across different subjects demonstrates the necessity of subject-aware strategy selection.

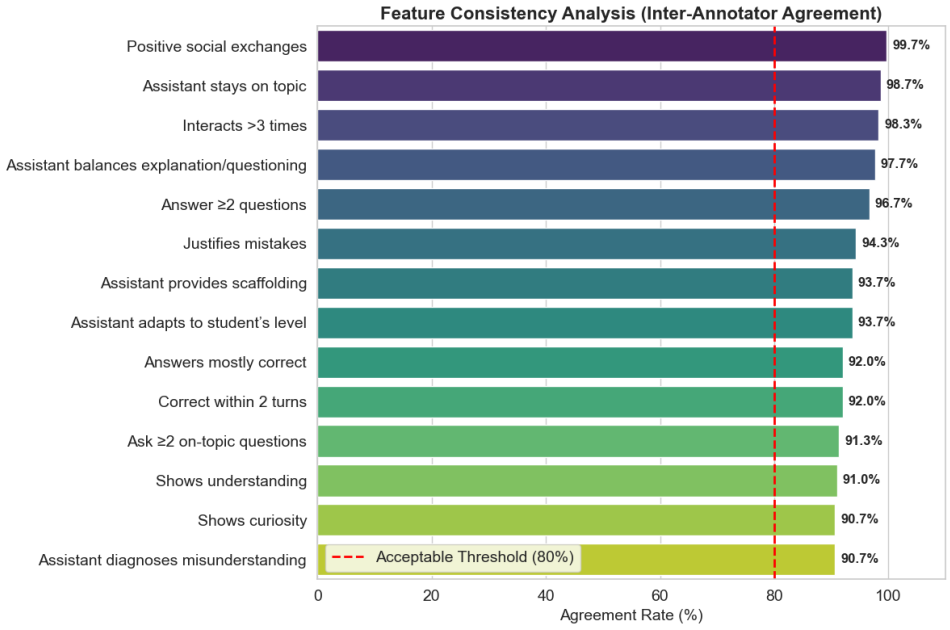
Finally, Figure 5.4 displays the average score across different tutor strategies and academic subjects. The results show that there is no "silver bullet" for all disciplines. For example, in English (Engels), the *Challenge* strategy achieves a higher average score (0.721). On the other hand, in Math (Wiskunde), the Feynman method outperforms the other tutors (0.705). This critical variance indicates that a static prompt is insufficient, thereby justifying the necessity of a Subject-aware prompt routing mechanism to dynamically map specific subjects to their optimal pedagogical strategies.

## 5.2 Consistency of Feedback Features

Since the LLM evaluator operates as a black box, the consistency of its output is highly important. To validate the reliability of our feature-based score mechanism, we evaluated the self-consistency across multiple evaluation sessions. Figure 5.5 illustrates the consistency rate for each pedagogical feature. By comparing the two settings, using three runs improves the overall consistency from 91.29% to 94.31%. This setting also ensures that the lowest individual feature consistency remains at 90.7%, compared to 82.0% in the single run. While multiple runs provide higher consistency, they also involve more computational power, API costs, and running time. In this thesis, we leveraged a three-run evaluation to balance inference time and the consistency of the feedback signal.




(a) Single Evaluation Run




(b) Three Evaluation Runs

Figure 5.5: Consistency of Feedback Features. The horizontal bars represent the agreement rate for each criterion. Figure (a) shows the consistency from a single evaluation run with an overall agreement of 91.29%. Figure (b) demonstrates the results after aggregating three independent evaluation runs (e.g., via majority vote). By utilizing multiple runs, the overall consistency improves from 91.29% to 94.31%, ensuring all evaluated features robustly surpass the 80% acceptable threshold (red dashed line) and providing a more stable feedback signal for router's training.

## 5.3 Result of Training

### 5.3.1 Overall Learning Convergence in Simulation

Figure 5.6 illustrates the training dynamics within the LLM-based simulation environment. At the beginning, the raise of average score rise swiftly, suggesting that the routing model rapidly adapts to the pedagogical environment and identifies effective strategies. Over the course of training, the model improves the average score from $0.646 \pm 0.152$ (initial 1,000 steps) to $0.671 \pm 0.137$ (final 1,000 steps), demonstrating the router's capability to optimize pedagogical prompt selection.

Although the absolute margin of improvement appears modest, this is primarily due to a natural ceiling effect introduced by the high quality of the base prompts. Because the prompt space consists of expert-crafted pedagogical strategies, even a random initial model yields a relatively high baseline score, thereby compressing the available margin for optimization.

Furthermore, the wide variance in the raw episodic score (ranging approximately from 0.2 to 0.8, as shown in the background of the figure) highlights the high level of irreducible noise inherent in the simulated students. This wide distribution mimics the unpredictable nature of real-world human interactions. The steady convergence of the smoothed score, despite this chaotic baseline, confirms that our router successfully learns a robust and stabilized strategy in a highly noisy environment.

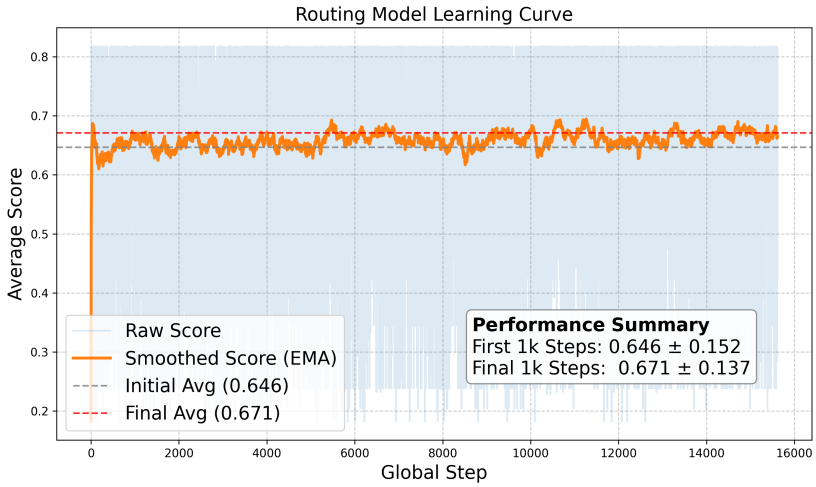



Figure 5.6: Learning curve in the simulated environment. The light blue background shows the raw episodic score, illustrating the high environmental noise, while the orange line represents the smoothed Exponential Moving Average (EMA) score. The router swiftly improves the average score at the beginning, and elevates the mean score and reduces the standard deviation at the end of training. Overall, these dynamics confirm that the routing model can effectively overcome environmental noise to consistently select optimal, expert-crafted strategies.

### 5.3.2 Decoupling of Subject Representations

To overcome the issue of highly uniform input representations observed in Figure 3.2, we introduced a trainable subject embedding layer into the model architecture. This addition enables the router to learn and apply distinct pedagogical strategies across different academic disciplines.

The pairwise cosine similarity of the learned subject embeddings is presented in Figure 5.7, revealing a well-separated and nearly orthogonal latent space between different subjects. By fusing the frozen textual input embeddings (representing the topic title) with these dynamic subject embeddings, the routing model can effectively process both the topic title and the subject id of each educational scenario.

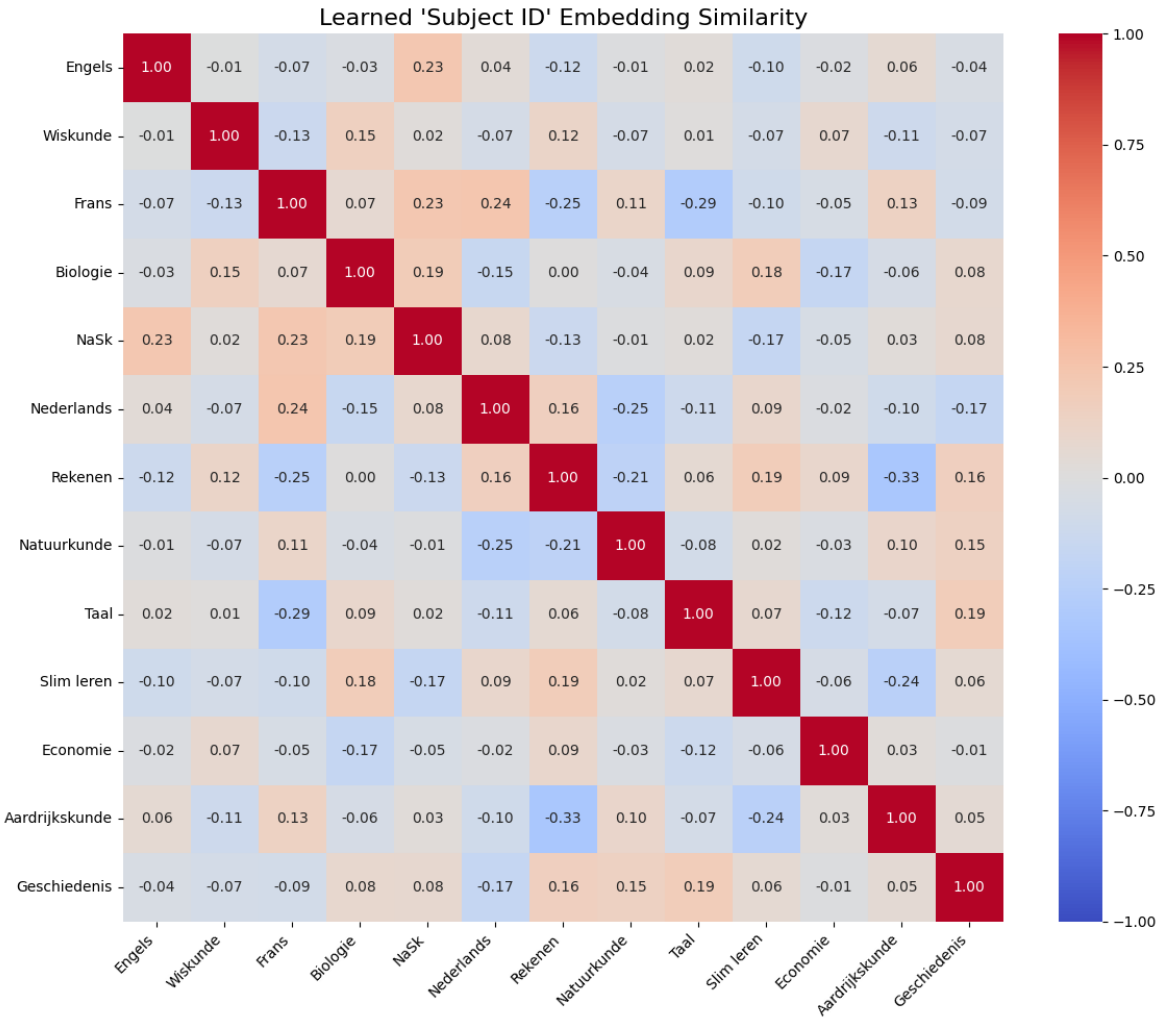



Figure 5.7: Cosine Similarity Heatmap of the Learned Subject Embeddings. In stark contrast to the collapsed semantic space of the pre-trained title embeddings (Figure 3.2), the learned subject embeddings exhibit near-orthogonality, with inter-subject similarities predominantly clustered around 0 (ranging from -0.33 to 0.24). This demonstrates that the trainable subject embedding layer successfully extracts highly distinct, decoupled representations for each academic discipline, effectively providing the policy network with clear subject anchors.

### 5.3.3 Score Stratification by Environment and Strategy

To further understand the model's training dynamics, we perform a stratified analysis of the learning curves across different student profiles and tutor strategies.

Figure 5.8 illustrates the learning curves by three simulated student profiles. As the overall learning curve, all learning curves rise rapidly at the beginning, suggesting that the routing model can swiftly adapt to different student types. Notably, while all curves exhibit high initial variance, the Unmotivated Student shows a steady, marginal improvement toward the end of training. This suggests that the router eventually discovers optimal strategies that can elicit slightly better learning quality, even in a challenging scenario.

Moreover, A significant score gap is observed between Motivated, Mediocre, and Unmotivated students throughout the training process. This gap highlights the high uncertainty within the educational environment; interactions with unmotivated or uncooperative students impose a lower theoretical bound on achievable scores, regardless of the tutor's proficiency. This score difference proves the validity of our LLM evaluator once again.

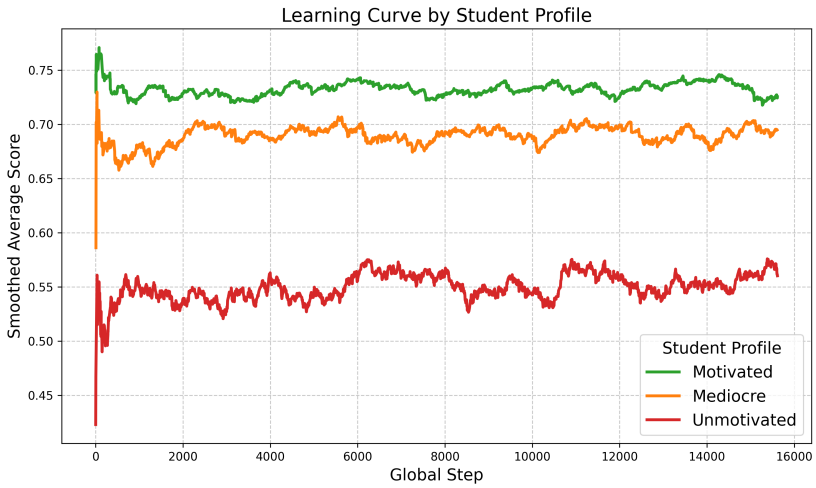



Figure 5.8: Stratified Learning Curve by Student Profile. The learning curves rise rapidly initially. The curve of Unmotivated Student shows margin improvement and stabilization in the end of training, indicating that the router can discover optimal strategies for handling a challenging scenario. Moreover, the figure illustrates the score gap imposed by different student levels, reflecting environmental complexity and the validity of our pedagogical scores.

To analyze different strategies, Figure 5.9 presents the score dynamics stratified by the top-performing tutor strategies. The initial phase fluctuates as the router explores the prompt space. Moreover, the high score at the beginning of the *Feynman* might be due to sampling coincidence, as *Feynman* luckily sampled the Motivated students more.

Furthermore, as training progresses, we observe a clear convergence where deep-reasoning strategies, specifically *Analytical_opt* and *Feynman*, emerge as the most robust performers. The narrowing variance across these strategies indicates that the router has

successfully transitioned from random exploration to an optimized pedagogical strategy, consistently selecting prompts that maximize long-term educational efficacy across diverse student states.

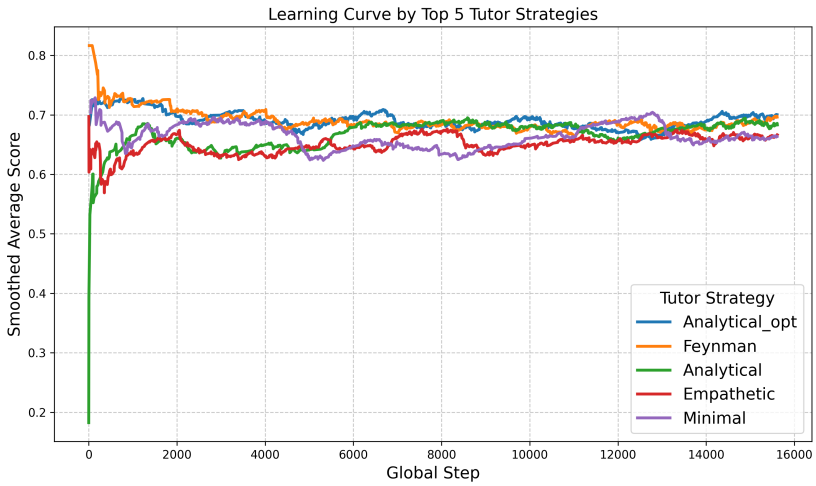


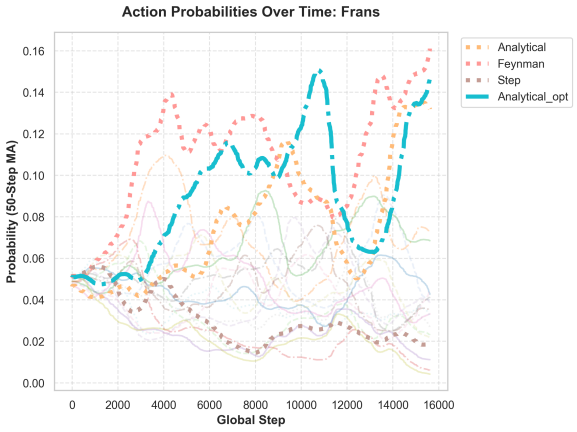

Figure 5.9: Stratified Learning Curve by Tutor Strategy. The variance in the initial training process suggests the exploration phase for the router. The high score prompts eventually dominate, indicating that the routing model can select optimal strategies.

### 5.3.4 Emergence of Subject-Specific Strategies

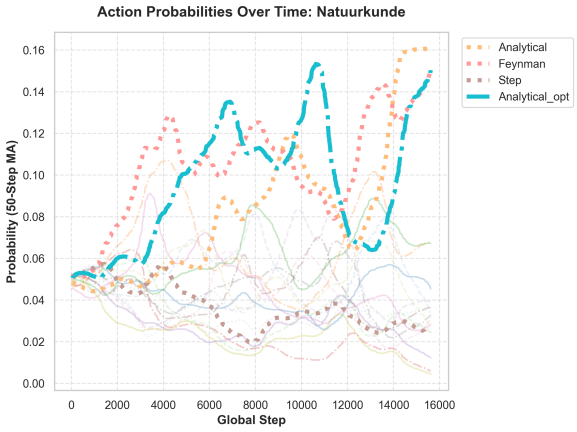
While the score stratification in Section 5.3.3 demonstrates the impact of the prompt space and environment, a true Contextual Bandit must demonstrate input-dependent behavioral shifts. To verify the Subject-aware Prompt Selection, we tracked the evolution of the prompt probability distribution across different subjects. Rather than exhibiting a learning curve, the router exhibits distinct, subject-aware strategy selections. Figure 5.10 highlights four representative subject cases that capture the spectrum of the router's behavior.

Figures 5.10a and 5.10b illustrate successful, distinct specializations. For French (Frans), the router rapidly identifies and commits to the *Feynman* strategy. Conversely, for Physics (Natuurkunde), the router distinctly shifts its preference toward the *Analytical* approach. This contrast definitively proves that the router avoids global mode-collapse and successfully maps pedagogical inputs to their respective strategies.

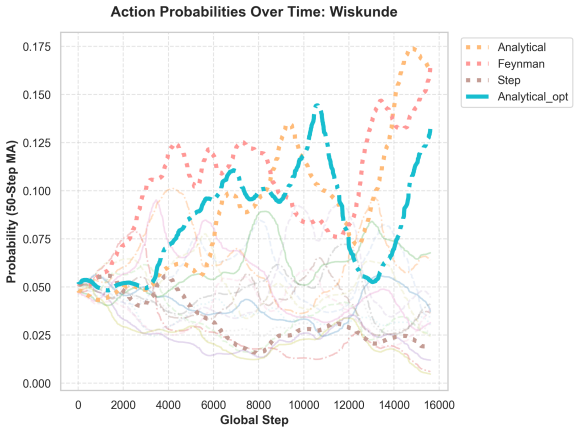
Figure 5.10c captures a highly competitive scenario in Mathematics (Wiskunde). The router initially defaults to the globally dominant *Feynman* strategy. However, as training progresses, a strategic crossover occurs around step 10,000, with the router rapidly increasing its preference for the *Analytical_opt* approach. This highlights the router's capacity to dynamically compare and shift between top-tier strategies as its representation of the input space matures.



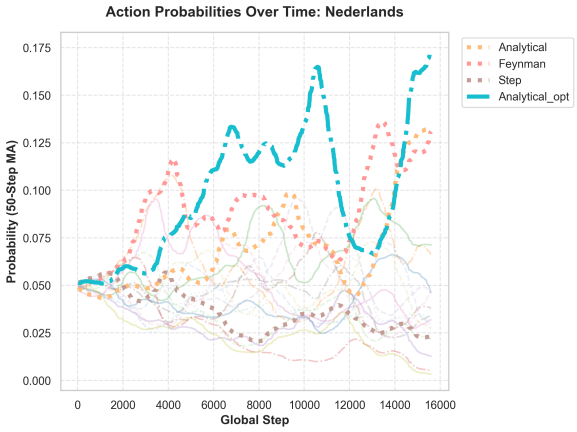

(a) French: Convergence to Feynman




(b) Physics: Convergence to Analytical




(c) Math: Competitive Crossover




(d) Dutch: Premature Convergence

Figure 5.10: Taxonomy of learning dynamics across different academic subjects. The key strategies discussed are highlighted: Feynman (pink dotted), *Analytical_opt* (teal dash-dot), *Analytical* (orange dotted), and *Step* (brown dotted). The router demonstrates subject-specific adaptations, ranging from confident convergence (a, b), dynamic strategic crossovers (c), and occasional convergence to local optima due to strong global priors (d). Those results suggest that our Subject-aware Prompt Routing framework can dynamically choose strategies based on different subjects during the training in the noisy environment.

Finally, Figure 5.10(d) presents a learning challenge observed in Dutch (Nederlands). While the empirical optimal strategy is *Step* (as will be shown in Section 5.3.5), its probability remains suppressed throughout training. Because Analytical_opt serves as a highly robust global prior, the router likely discovered that it yields "good enough" score early on. Consequently, it heavily exploited this known strategy, failing to sufficiently explore the *Step* prompt, a textbook example of premature convergence to a local optimum.

### 5.3.5 Empirical Alignment of the Learned Policy

To quantitatively validate whether the router successfully discovered the optimal Input-Prompt mapping, we conducted an empirical alignment analysis. We compared the

Table 5.1: Comparison of Empirically Optimal Strategies vs. Model-Predicted Strategies across Subjects in simulation. The metric "Gap" is the regret between the theoretical maximum score and the model's chosen prompt. The alignment rate achieves 69.2%, suggesting that the router converges to a near-optimal solution that can handle most scenarios.

| Subject | Empirical Best Prompt | Mean Score | Model Predicted Prompt | Predicted Mean Score | Gap |
|---|---|---|---|---|---|
| Geography | **Analytical_opt** | 0.712 | **Analytical_opt** | 0.712 | **0.000** |
| Biology | Analytical | 0.696 | Analytical_opt | 0.674 | 0.022 |
| Economics | **Analytical_opt** | 0.698 | **Analytical_opt** | 0.698 | **0.000** |
| English | Challenge | 0.721 | Analytical_opt | 0.716 | 0.005 |
| French | **Feynman** | 0.722 | **Feynman** | 0.722 | **0.000** |
| History | Best | 0.723 | Analytical_opt | 0.710 | 0.013 |
| PhysChem | **Analytical_opt** | 0.703 | **Analytical_opt** | 0.703 | **0.000** |
| Physics | **Analytical** | 0.727 | **Analytical** | 0.727 | **0.000** |
| Dutch | Step | 0.693 | Analytical_opt | 0.654 | 0.039 |
| Calculus | **Analytical_opt** | 0.685 | **Analytical_opt** | 0.685 | **0.000** |
| Learning Skills | Emotion_First | 0.722 | Analytical_opt | 0.709 | 0.012 |
| Language | **Analytical_opt** | 0.713 | **Analytical_opt** | 0.713 | **0.000** |
| Mathematics | **Feynman** | 0.705 | **Feynman** | 0.705 | **0.000** |
| **Strategy Alignment Rate (Gap $\leq$ 0.01): 69.2%** | | | | | |

router's final predicted prompt against the empirical ground truth. The empirical best prompt for each subject was determined by calculating the maximum average episodic score across all available interaction logs for that domain.

Table 5.1 presents this strategy comparison. The "Gap" metric represents the empirical regret, the difference in expected score between the theoretical best strategy and the strategy ultimately selected by the model.

The results demonstrate a degree of strategy alignment. In 8 out of the 13 evaluated subjects, the router achieved a perfect match (Gap = 0.000), successfully identifying and adopting the statistically optimal pedagogical strategy. Notably, in subjects with highly competitive prompt spaces like Mathematics (Wiskunde), the model successfully navigated the strategic crossover (as seen in Figure 5.10c) to ultimately converge on the empirical optimum, the *Feynman* strategy.

In subjects where a gap exists, the regret margins remain remarkably narrow (averaging under 0.02 among non-matching subjects). For instance, in Biology (Biologie), the empirical optimum was the strictly *Analytical* strategy (0.696). Although the model ultimately prioritized the globally dominant *Analytical_opt* approach (0.674) yielding a minor gap of 0.022, this behavior is highly interpretable. It indicates that when faced with closely performing strategies, the router occasionally exhibits premature convergence to a strong, high-performing global prior (*Analytical_opt*) rather than risking exploration for a marginally better local optimum. Ultimately, this table proves that the Subject-aware Prompt Routing framework successfully converges to near-optimal, Subject-aware instructional strategies.

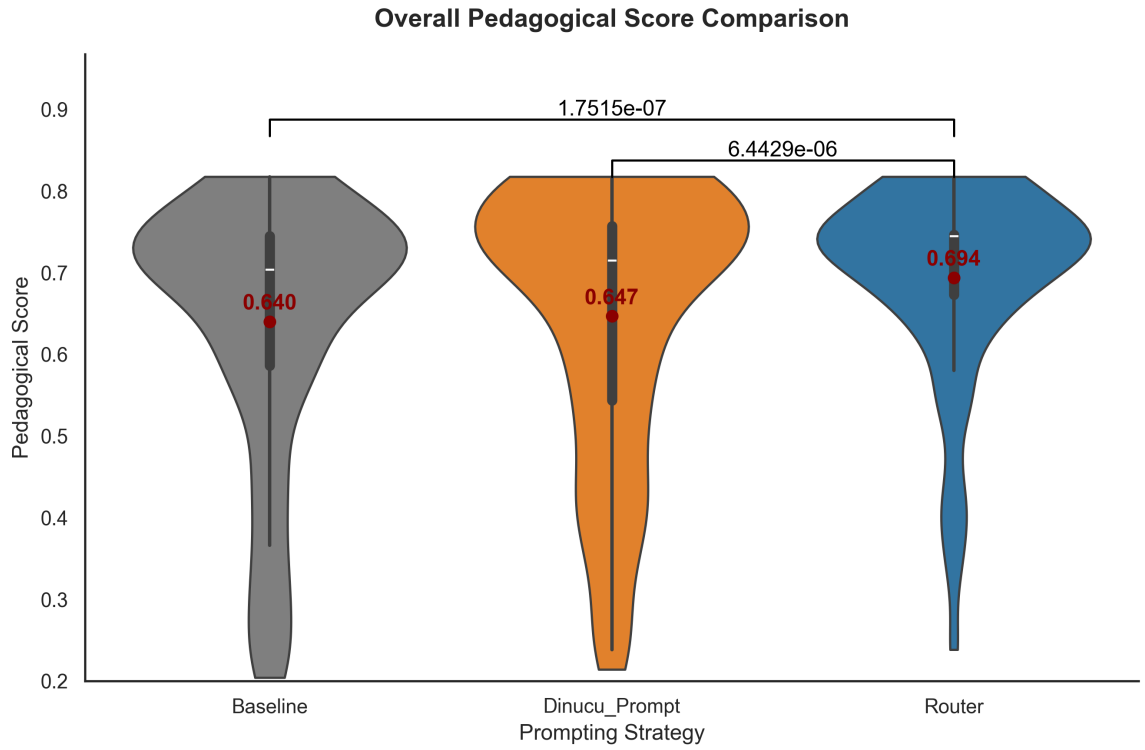



Figure 5.11: Overall pedagogical score distribution across 390 simulated scenarios. The router dynamic prompt selection not only achieves the highest median score but significantly raises the performance floor, eliminating the severe pedagogical failures (scores below 0.4) observed in static baselines.

## 5.4 Simulated Benchmark: Comparative Analysis

### 5.4.1 Overall Performance and Statistical Significance

To provide a rigorous evaluation of the pedagogical efficacy, Figure 5.11 presents the full distribution of the expected scores across all 390 test scenarios (comprising 13 subjects, 10 topics, and 3 student profiles).

The visualization reveals a profound structural advantage of the Subject-aware Prompt Routing. While the router achieved the highest aggregate mean score of 0.694 (outperforming both the state-of-the-art Dinucu_Prompt at 0.647 and the Baseline at 0.640), the plot illustrates how this improvement is distributed. The routing model significantly raises the performance floor and compresses the variance. Most importantly, it effectively prevents severe pedagogical failures, the heavy tail of low-scoring outliers clustering between 0.2 and 0.4 seen in both static approaches.

To confirm that this performance gain is not due to random variance, independent two-sample t-tests were conducted. The results demonstrate that the superiority of the router is statistically significant when compared to both baselines ($p < 0.001$). This confirms that dynamic, subject-aware strategy selection fundamentally surpasses single, globally optimized static prompts in overall stability and effectiveness.

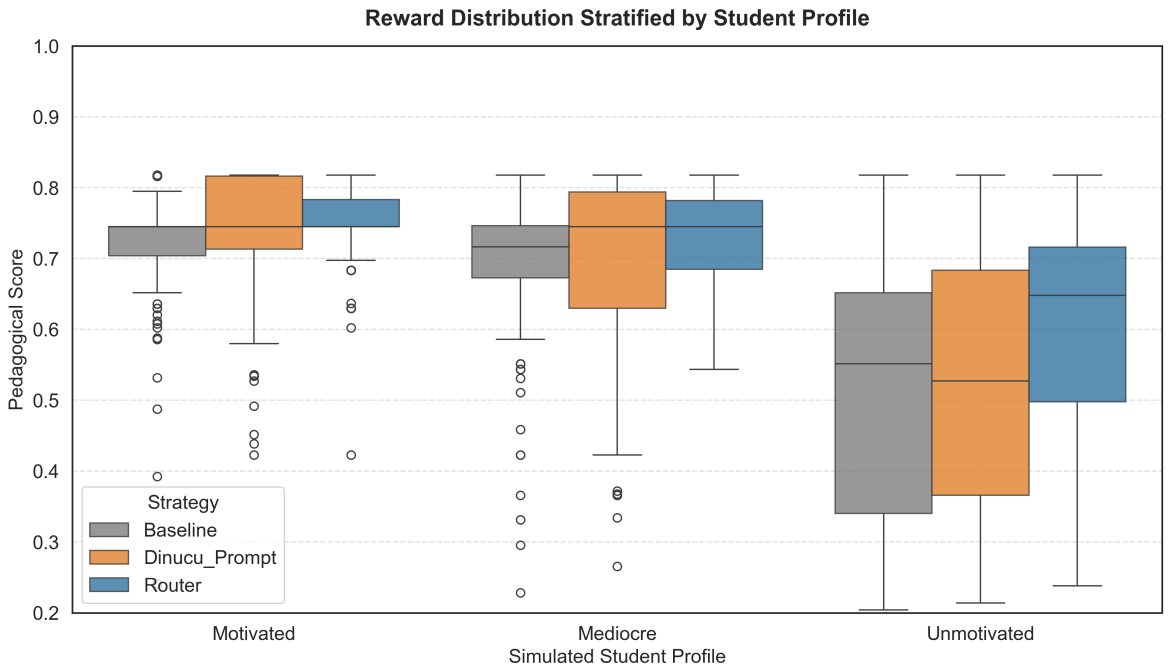



Figure 5.12: score distribution stratified by simulated student profile. While static baselines experience severe variance and plummeting medians when facing uncooperative Unmotivated students, the router demonstrates superior robustness by maintaining a tight performance distribution.

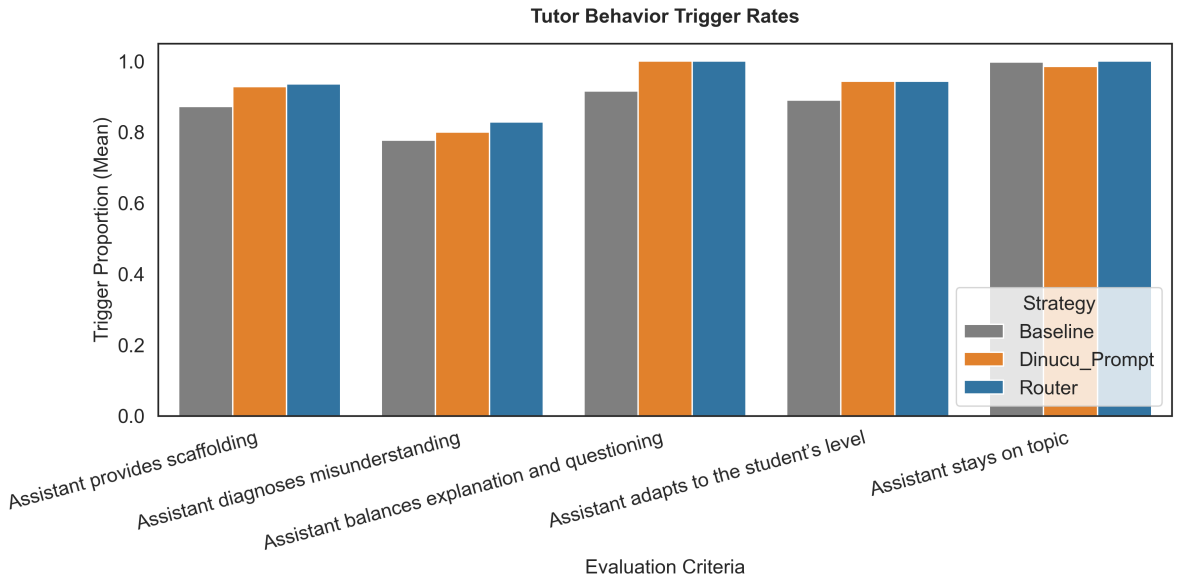
### 5.4.2 Robustness Across Student Profiles

A hallmark of an effective tutor is the ability to adapt to struggling or unmotivated learners. As shown in Figure 5.12, when interacting with Motivated and Mediocre students, all three strategies maintain relatively high medians, though the router consistently exhibits less variance and fewer low-scoring outliers.

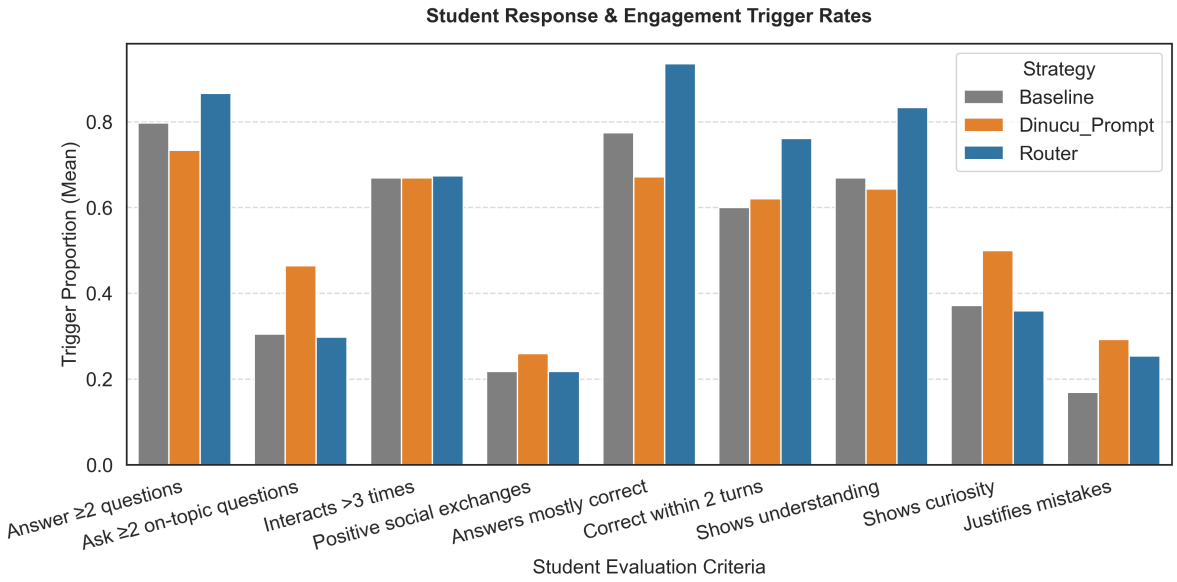
However, the critical divergence occurs when facing the Unmotivated student profile. Both static prompts experience a severe performance collapse, characterized by lower medians and massive variance (with interquartile ranges stretching down to 0.35). Their rigid instructions fail to systematically re-engage distracted learners. In contrast, the router maintains remarkably high score stability. It preserves a median score near 0.65 and compresses the degradation margin. This proves that the Subject-aware Prompt Routing effectively acts as a cognitive safety net, significantly enhancing the system's pedagogical robustness in highly challenging environments.

### 5.4.3 Behavioral Mechanisms of Success

To unpack the "black box" of the router's higher scores, we analyzed the trigger rates of specific evaluation criteria during the benchmark, divided into Tutor behaviors and Student responses (Figure 5.13).




(a) Tutor Behavior Trigger Rates




(b) Student Response & Engagement Trigger Rates

Figure 5.13: Detailed criteria analysis. (a) The router outperforms baselines in providing scaffolding and adapting to the student's level. (b) Consequently, this adaptive tutoring leads to higher student engagement, characterized by answering more questions and achieving higher correctness.

As shown in Figure 5.13(a), the router consistently elicited a higher proportion of desirable teaching behaviors from the LLM, notably excelling in "Assistant provides scaffolding" and "Assistant adapts to the student's level." More importantly, Figure 5.13(b) demonstrates the direct impact of these interventions: the simulated students engaged more deeply ("Answer $\geq 2$ questions") and performed better ("Answers mostly correct"). This empirically validates that the router did not merely learn to "game" the score function, but rather learned to deploy instructional techniques that genuinely

foster active student participation.

## 5.5 Real-World Deployment and A/B Testing

While Simulation results (Section 4.4) establish the theoretical superiority of the prompt in an idealized environment, they also represent a 'closed-world' bias. In this section, we investigates whether these learned strategies can survive the 'noise' and unpredictable behaviors of real-world students (RQ3). To answer our third research question, we performed a live A/B test with $N = 656$ interactions with 359 Dutch high-school students.

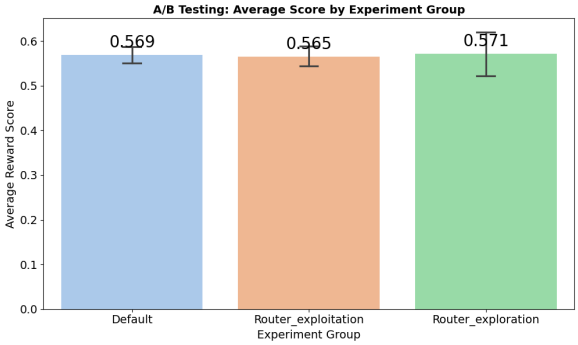
To evaluate the framework in a production environment, the routing model was pretrained in simulation for 2,000 steps to facilitate sim-to-real knowledge transfer. This online deployment allows us to observe whether the pedagogical preferences optimized in simulation align with real-world user behaviors and preferences.

### 5.5.1 Pedagogical Quality and Robustness

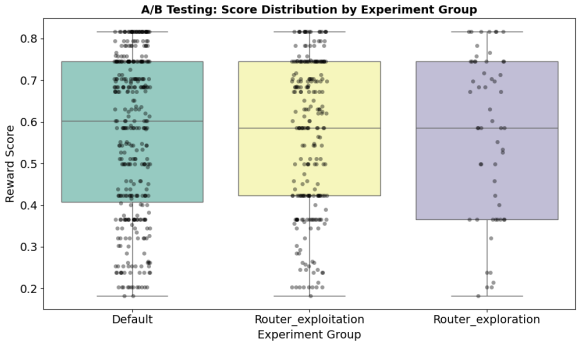
As shown in Figure 5.14a, the average scores of the control group (Default) and the variation group (Router_exploitation) are comparable. However, Figure 5.14b illustrates that the Router_exploitation group exhibits a higher first quartile (Q1) compared to the default group. This indicates that the bottom 25% of sessions in the Router group achieved improved score scores, suggesting enhanced pedagogical consistency and a more robust "floor" for teaching quality during the online transfer process.

The pedagogical quality of the Router_exploitation group (0.565) is statistically indistinguishable from the Default baseline (0.569, $p = 0.6349$). This high degree of alignment underscores the stability of our Router framework, ensuring that the transition from simulation to real-world deployment does not compromise instructional standards. In the final 7 days of deployment, the router achieved an average score of 0.612 ($n = 35$) compared to the baseline's 0.596 ($n = 51$), although constrained by sample size ($p = 0.461$).

Furthermore, the slightly higher average score in the Router_exploration group suggests that the policy model still has potential for growth as more real-world interaction data is incorporated into the learning loop.




(a) Average Scores

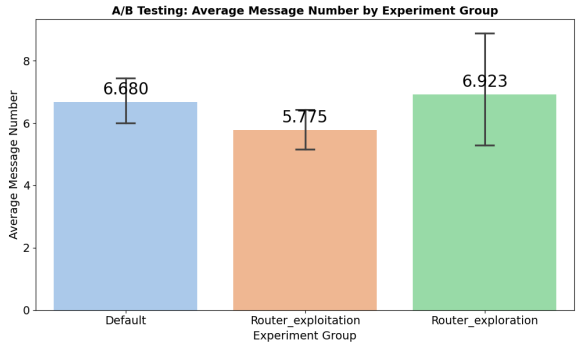



(b) Score Density and Distribution

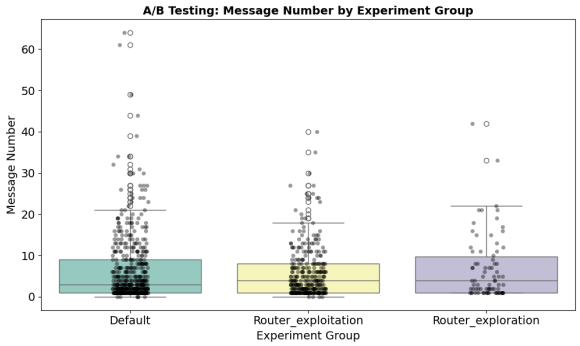
Figure 5.14: A/B Testing results comparing the Static Baseline (Default) with the Router-selected strategies. (a) shows the robustness of the router in maintaining baseline performance; (b) illustrates the underlying distribution and variance across groups. The Router_exploitation shows a higher first quartile but lower median and average, indicating that the router is robust in the real-world environment, but still has space to improve.

### 5.5.2 Analysis of Interaction Dynamics and Efficiency

The observed reduction in average message count (6.68 vs. 5.775) suggests that the Router-selected strategies provide more precise pedagogical guidance, effectively minimizing redundant conversational loops often found in the baseline. Simultaneously, the shifts in the distribution as shown in Figure 5.15b suggest that while the router maintains high pedagogical quality, it actively seeks to optimize the efficiency. A detailed examination of these dynamics, particularly in longer sessions, reveals how the router prunes redundant conversational loops.




(a) Average Message Number




(b) Message Number Distribution

Figure 5.15: Comparison of interaction efficiency across experiment groups. (a) shows the reduction in average message count for the router group, suggesting that the router improves pedagogical efficiency. (b) illustrates the distribution and pruning of redundant conversational paths. The narrow distribution of router_exploitation shows that the router can not only reduce redundant conversation but also the early drop-off case.

While the total dataset shows a visible difference in message counts between the control and variant groups, the associated p-value ($p = 0.63$) indicates that the overall result

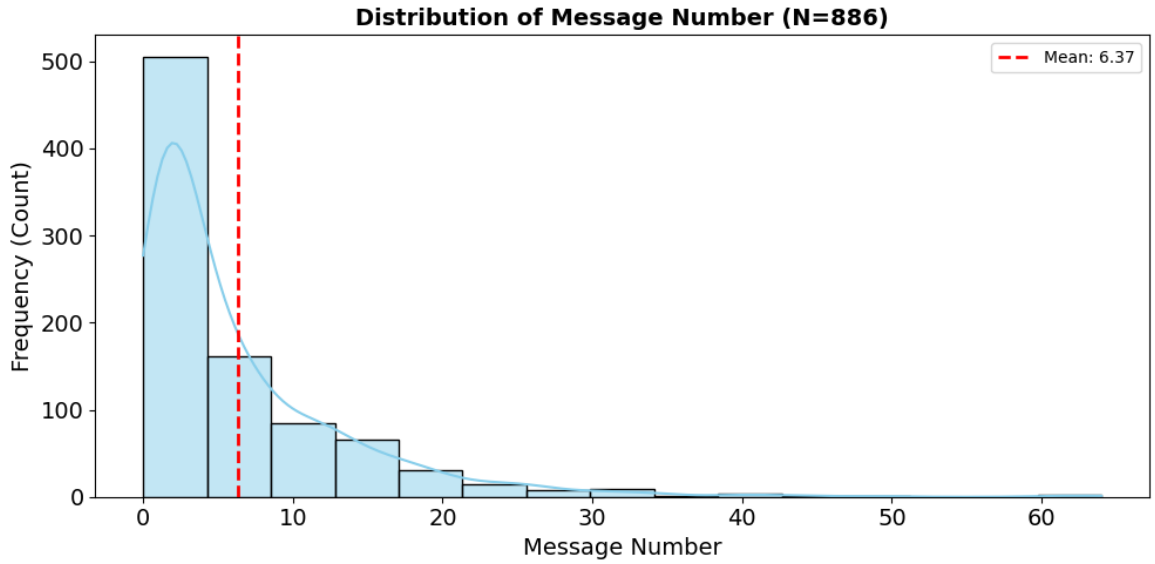
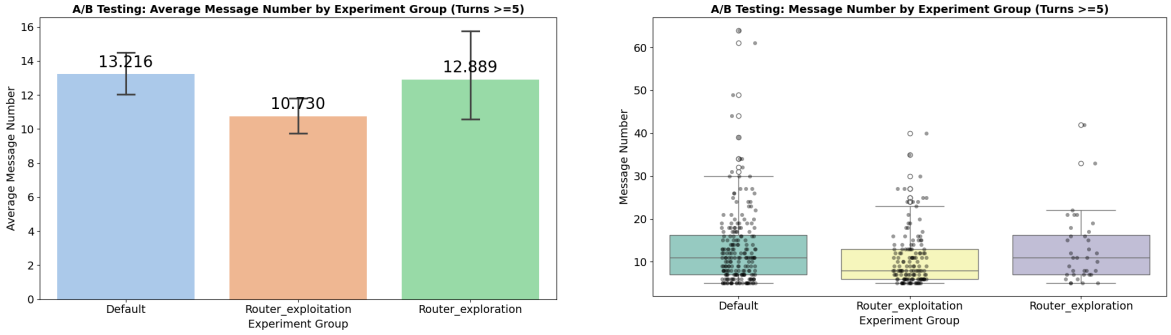



Figure 5.16: Overall distribution of conversational turns. The dataset exhibits a high concentration of sessions with 1 to 3 messages.

is not statistically significant. This prompted an investigation into the session length distribution, as shown in Figure 5.16, which reveals a high concentration of sessions with only 1 to 3 messages.

From a pedagogical perspective, an AI tutor typically requires approximately five turns to effectively introduce a topic, diagnose student misunderstandings, and facilitate scaffolding. Consequently, these short-form interactions function as "cold-start" noise that obscures the impact of specialized teaching strategies. By isolating "Substantial Interactions" where the message count is $\geq 5$, a different trend emerges.

As illustrated in Figures 5.17a and 5.17b, both the average (around 3 turns less) and median message counts are lower in the Router_exploitation group, suggesting superior instructional efficiency. Furthermore, for sessions with turns $\geq 5$, the p-value drops to 0.0057 ($p < 0.05$), confirming a statistically significant difference between the static baseline (Default) and the dynamic Router-optimized strategy. This demonstrates that in deep pedagogical exchanges, the router is capable of guiding students to learning objectives through more concise conversational paths.




(a) Average Message Number

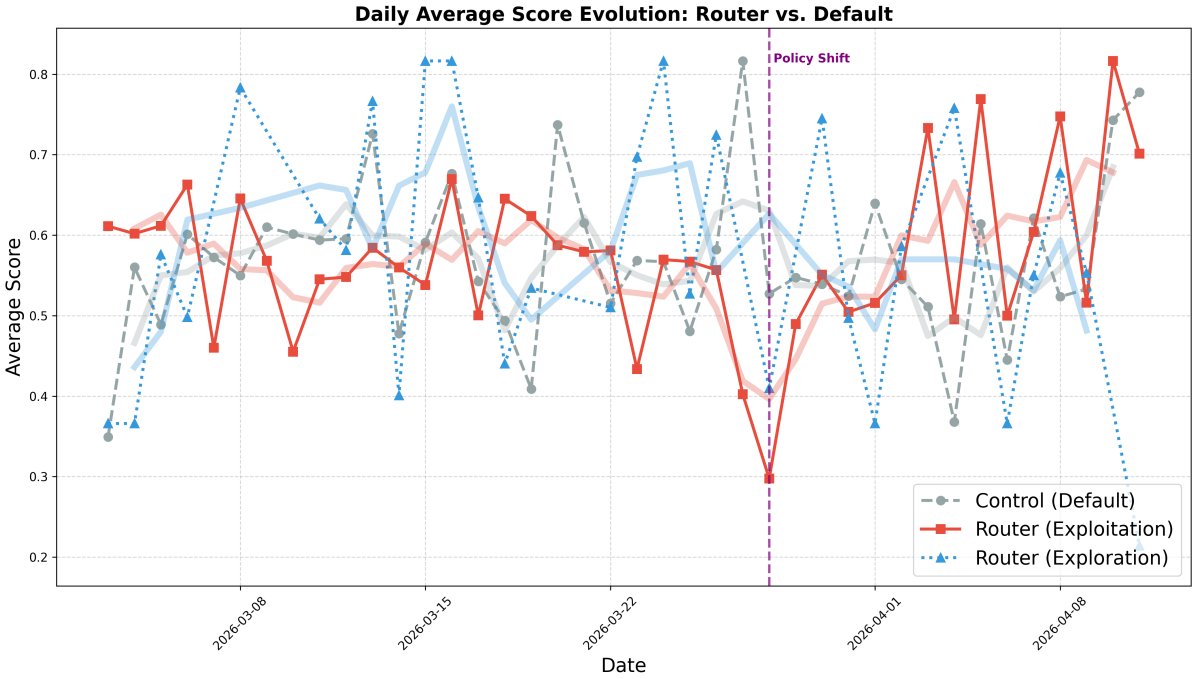
(b) Message Number Distribution

Figure 5.17: Comparison of interaction length for substantial interactions (Turns $\geq$ 5). The Router_exploitation reduces around 3 turns in the conversation. These results further confirm that the router can improve pedagogical efficiency with statistical significance without cold-start noise.

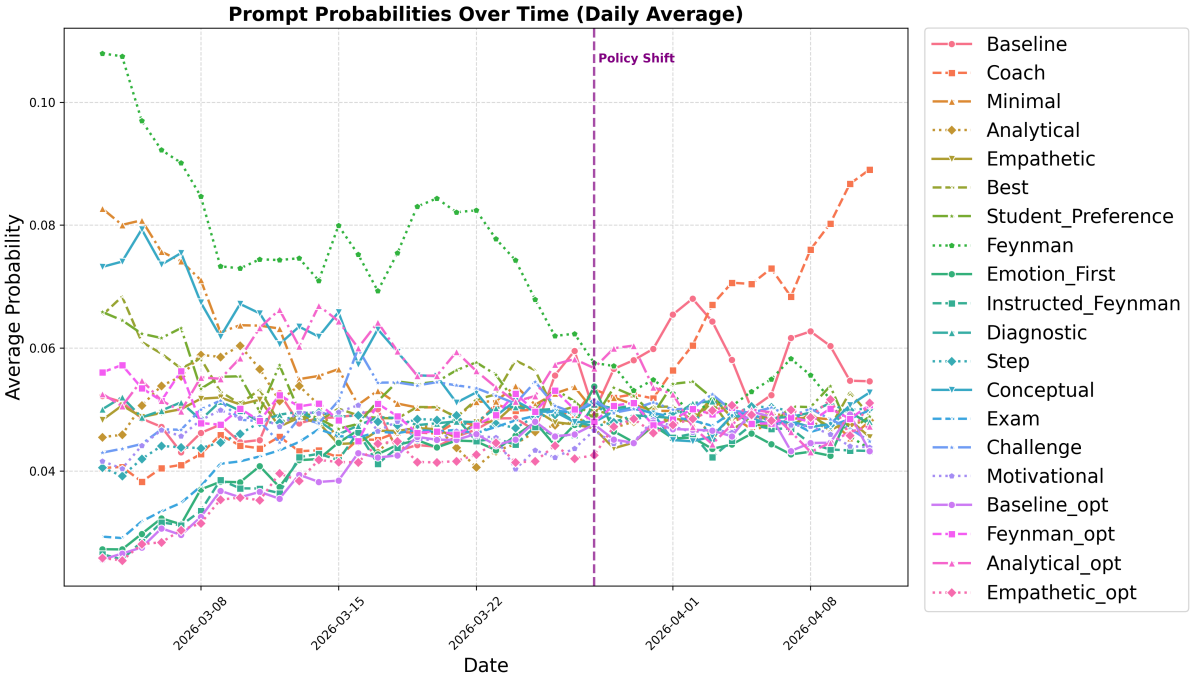
### 5.5.3 Strategy Dynamics and Learning Evolution

The real-world adaptation of the router is characterized by a distinctive learning pivot observed around March 28th. As illustrated in Figure 5.18a, the Router (Exploitation) group experienced a temporary performance dip on this date, followed by a robust and sustained recovery that eventually outperformed the static baseline.

This transition is explained by the action probability shifts shown in Figure 5.18b. Prior to the "Policy Shift" marker, the router relied heavily on the *Feynman* strategy, a preference inherited from the simulation environment. However, the drop in score triggered a critical realignment: the router aggressively suppressed the *Feynman* probability while simultaneously reinforcing the *Coach* and *Baseline* strategies. This crossover marks the moment the model synchronized with actual student feedback, effectively "unlearning" simulation biases to embrace more effective instructional techniques. The subsequent upward trend in score confirms the successful convergence of this real-world optimized strategy.




(a) Daily Average Score Evolution




(b) Action Probabilities Over Time (Daily Average)

Figure 5.18: Evolution of performance and prompt selection. The vertical dashed line marks the "Policy Shift" around March 28th. (a) shows a performance ascent after the policy shift, indicating that the router starts to exploit the optimal strategy, such as *Coach* strategy. (b) illustrates a crossover in the policy shift, where the router transitioned from simulation-preferred strategies (*Feynman*) to real-world-optimized scaffolding (*Coach*), leading to superior score stability and improvement.

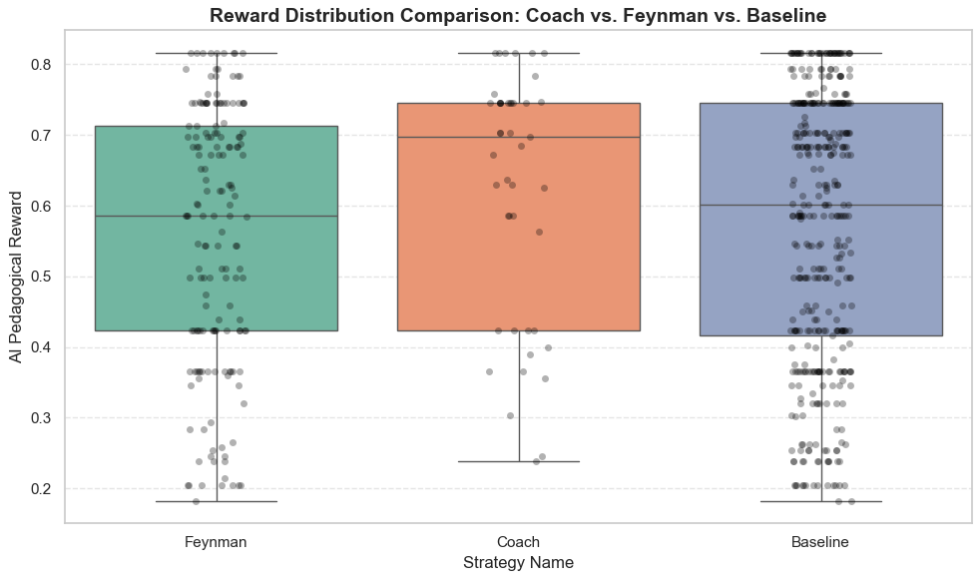



Figure 5.19: Statistical Significance and Performance Metrics of Teaching Strategies. The comparison shows that the high-scaffolding and supportive *Coach* strategy is significantly favored by real-world students, outperforming the deep-reasoning *Feynman* strategy that was preferred in the simulation.

While the global score average exhibited high variance during the adaptation phase, a focused analysis of the individual strategies reveals a clear hierarchical performance gap. As shown in Figure 5.19, the *Coach* strategy, which the model increasingly favored post-March 28th—demonstrated superior pedagogical effectiveness compared to both the Expert *Baseline* and the simulation-favored *Feynman* strategy.

Table 5.2 summarizes the statistical comparison between these core strategies. Notably, the *Coach* strategy achieved a significantly higher mean score (0.622) than *Feynman* (0.558) with a p-value of 0.0253 ($p < 0.05$). This empirical evidence confirms that the model's strategic shift was driven by its detection of superior student feedback for scaffolding-based interactions. Furthermore, it is worth noting that the *Coach* prompt functions as an LLM-generated variant built upon the foundational *Baseline* mechanics. Despite this, *Coach* outperformed the expert-curated *Baseline* with marginal significance ($p = 0.0675$), justifying the router's transition toward dynamic scaffolding over static heuristics. This result also proves that prompt-based pedagogical enhancements can effectively surpass traditional baseline limits.

Moreover, the result of live deployment highlights the different preferences between simulation students and real students. In the simulation, *Feynman* and *Analytical_opt* are preferred strategies, both emphasizing deeper reasoning and high cognitive demands. This strategy might lead to cognitive overload and frustration for real students. On the other hand, in the real-world environment, students benefit from high scaffolding and supportive strategies, such as *Coach* prompt. The difference suggests the challenge of an authentic simulation, since the LLM-based students perform more engagingly, even with the injection of distracted inner monologues.

Table 5.2: Statistical Significance and Performance Metrics of Teaching Strategies. The comparison shows a higher scaffolding and supportive strategy *Coach* is favored and benefits by real students rather than the simulation-preferred high reasoning *Feynman*.

| **Strategy Name** | **Mean score** | **Std. Dev.** | **Sample Size ($n$)** |
|---|---|---|---|
| Baseline (Expert) | 0.568 | 0.190 | 389 |
| Coach (Scaffolding) | **0.622** | 0.174 | 45 |
| Feynman (Explanation) | 0.558 | 0.185 | 164 |
| **Statistical Significance (Mann-Whitney U Test)** | | | |
| Coach vs. Feynman | $p = 0.0253$ (**Significant**) | | |
| Coach vs. Baseline | $p = 0.0675$ (Marginally Significant) | | |
| Feynman vs. Baseline | $p = 0.4489$ (Not Significant) | | |

### 5.5.4 Subject-Specific Evolution and Strategy Alignment

While the global trend indicates a systemic shift toward scaffolding, the router's adaptability is further evidenced by its subject-specific trajectories. As shown in Figure 5.20, the subject-aware embeddings allow the model to maintain strategies across different academic fields. The Table 5.3 shows the comparison between the empirical optimal and the final model-predicted prompt during live deployment.

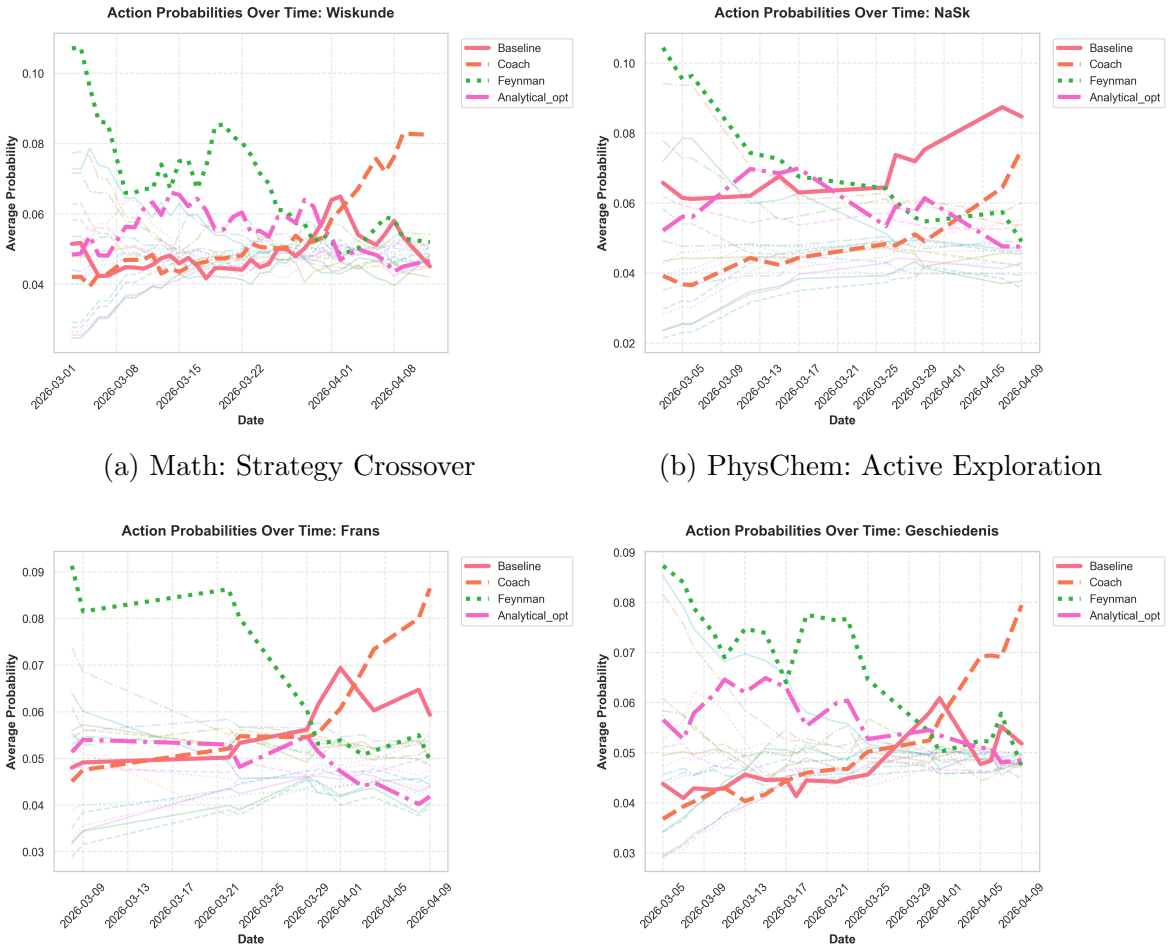
The subject-specific trajectories in Figure 5.20 reveal three distinct adaptation mechanisms. First, in high-volume subjects like Math (Wiskunde, Figure 5.20a), the router demonstrates rapid strategy recalibration. The clear crossover between *Feynman* and *Coach* probabilities around late March suggests that the router successfully identified and suppressed simulation-induced biases in favor of empirically superior scaffolding techniques. However, in the Table 5.3, the *Feynman* is marginally superior to the *Coach* (with a negligible gap of 0.008), suggesting that both strategies are competitive in math. Moreover, this minor discrepancy implies that the model's final convergence toward *Coach* is heavily influenced by the global prior established across the majority of the dataset, illustrating that the framework balances local domain tuning against a highly robust, cross-subject strategy.

Second, for subjects with sparser interaction data, such as French (Frans, Figure 5.20c) and History (Geschiedenis, Figure 5.20d), the strategy evolution mirrors the global trend toward *Coach*. This synchronization highlights the effectiveness of the subject-aware embedding framework, which enables implicit knowledge transfer. By leveraging pedagogical insights gained from high-traffic domains, the router can optimize teaching strategies even in data-poor environments. This cross-subject alignment is further verified in Table 5.3, where *Coach* consistently emerges as both the empirical optimum and the model's selected choice across all low-volume subjects that reached the evaluation threshold ($n \geq 5$).

Third, the different trend in PhysChem (NaSk, Figure 5.20b), where the *Baseline* remains dominant for a longer period, underscores the router's subject-specific sensitivity. Unlike the generic shift observed elsewhere, the model maintains a higher exploration rate for PhysChem, reflecting the unique complexity of science-based interactions. This deliberate exploration is echoed in Table 5.3, where the alignment gap for PhysChem

remains pronounced (0.189) despite a relatively large sample size ($n = 16$). This confirms that the subject-aware routing framework successfully insulates unique domains, preventing them from being overwhelmed by the global majority preference.

Overall, as quantified in Table 5.3, this resulted in an 80% global alignment rate for high-confidence subjects ($n \geq 5$), confirming that the subject-aware routing framework successfully balances global pedagogical standards with local domain requirements. However, half of the subjects consist of sample sizes below 5, suggesting the limitation in the real-world: data sparsity and imbalanced user interaction distributions across subjects.




(a) Math: Strategy Crossover

(b) PhysChem: Active Exploration

(c) French: Rapid Convergence

(d) History: Early-stage Evolution

Figure 5.20: Dynamic evolution of action probabilities during real-world deployment. (a) Math exhibits a clear strategic crossover where the model overrides simulation priors (Feynman) in favor of the Coach strategy. (b) In PhysChem, the router maintains the Baseline while actively exploring the rising *Coach* potential. (c) French demonstrates rapid convergence enabled by cross-subject knowledge transfer. (d) Even in subjects with sparse data like History, a consistent trend toward pedagogical scaffolding (*Coach*) is observed, aligning with the global policy shift. These results demonstrate that the router exhibits subject-specific decision-making, while also revealing how a globally dominant strategy can influence data-sparse domains.

Table 5.3: Live Deployment: Comparison of Empirically Optimal Strategies vs. Model-Predicted Strategies. The 'Gap' represents the empirical regret. Predicted Score is marked as *NaN* where interaction samples $n < 5$, reflecting the early-stage exploration and data sparsity.

| Subject | Empirical Best | Max Score | Model Predicted | Pred. Score | Sample | Conf. | Gap |
|---|---|---|---|---|---|---|---|
| Math | Feynman | 0.543 | Coach | 0.535 | 12 | 0.082 | **0.008** |
| French | Coach | 0.656 | Coach | 0.656 | 6 | 0.086 | **0.000** |
| Geography | Coach | 0.737 | Coach | 0.737 | 5 | 0.076 | **0.000** |
| Biology | Coach | 0.628 | Coach | 0.628 | 10 | 0.089 | **0.000** |
| PhysChem | Feynman | 0.644 | Baseline | 0.455 | 16 | 0.085 | 0.189 |
| History | Baseline | 0.612 | Coach | NaN | 4 | 0.079 | NaN |
| Dutch | Baseline | 0.638 | Coach | NaN | 3 | 0.074 | NaN |
| Physics | Baseline | 0.458 | Coach | NaN | 2 | 0.076 | NaN |
| English | Baseline | 0.559 | Coach | NaN | 2 | 0.077 | NaN |
| Economy | Feynman | 0.697 | Coach | NaN | 1 | 0.080 | NaN |
| **Overall Strategy Alignment Rate (for $n \geq 5$): 80.0%** | | | | | | | |

### 5.5.5 Sim-to-Real Trajectory and Bias Correction

To investigate the model's adaptation during the transition from synthetic to live environments, we performed a compared trajectory between the live deployment path and the continued simulation path. Figure 5.21 illustrates the dynamic difference between synthetic and real-world environments. While a visible "reality penalty" can be observed, the initial convergence is rapid when the routing model switches to the real deployment, indicating that the router performs excellent adaptation and robustness, unlearning the simulation bias. However, the one-month training corresponds to around 1000 steps in training, reflecting the data sparsity in the real world.

Furthermore, the alignment by training steps reveals a decisive strategy realignment. As illustrated in Figure 5.22, the router proactively suppressed the *Feynman* strategy, which was over-optimized in simulation, and favored the *Coach* strategy during the live deployment. This comparison once again confirms that a scaffolding and supportive strategy is preferred in real students, compared to a deeper reasoning method.

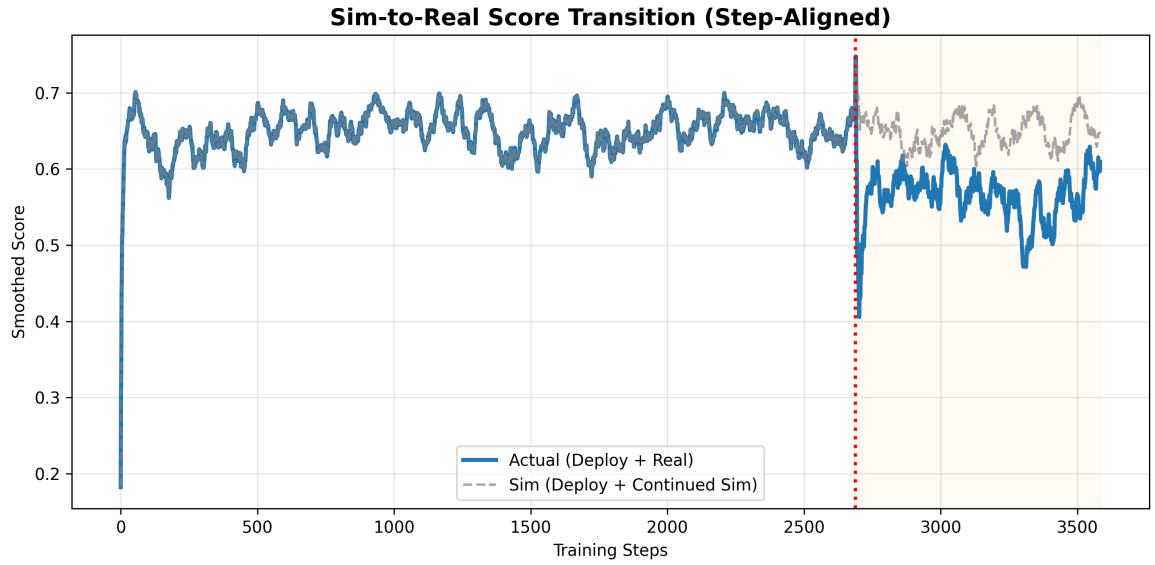



Figure 5.21: Sim-to-Real Score Transition (Step-Aligned). The red dotted line denotes the start of live deployment. While the actual score (blue) reflects the inherent complexity and penalty of human interaction, the router shows a high adaptivity with a rapid convergence when it switches to the real environment.

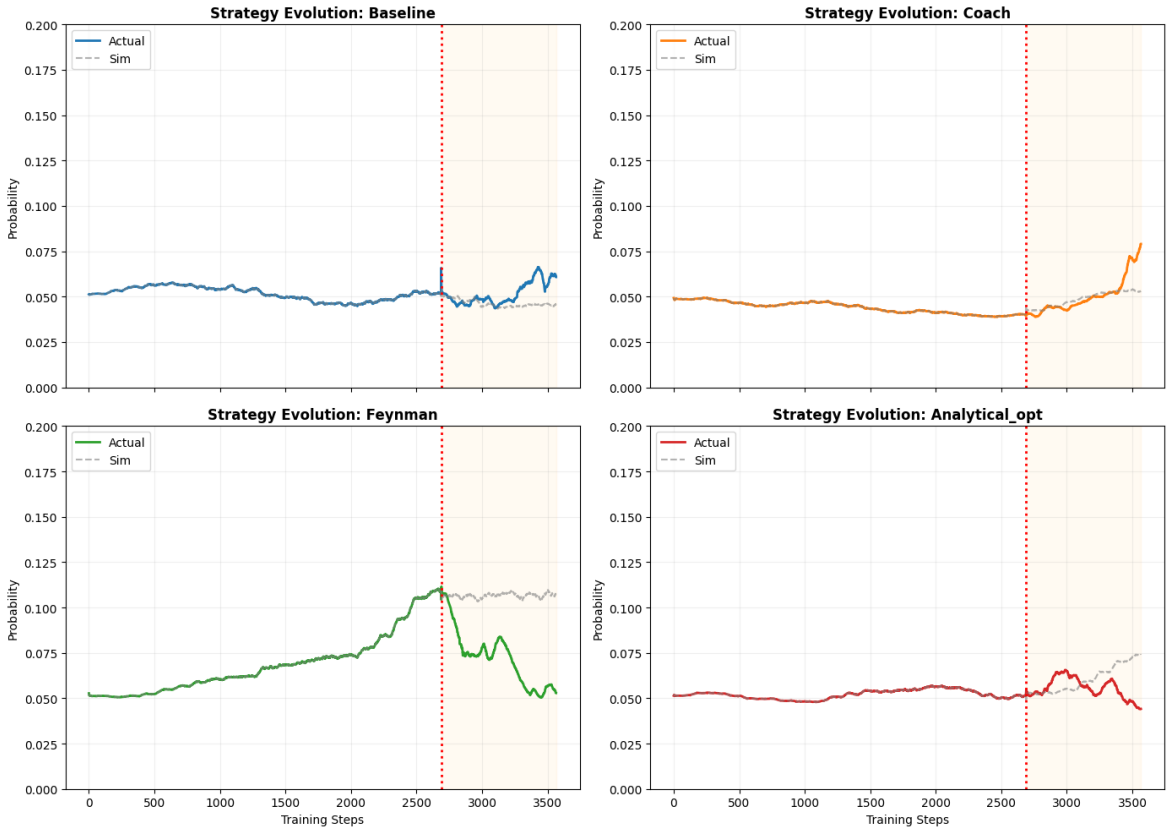
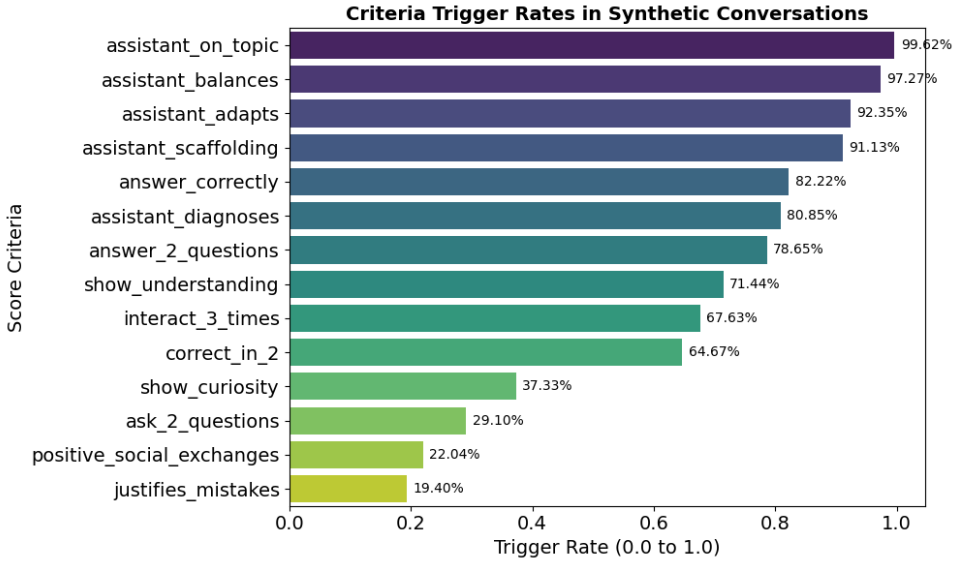


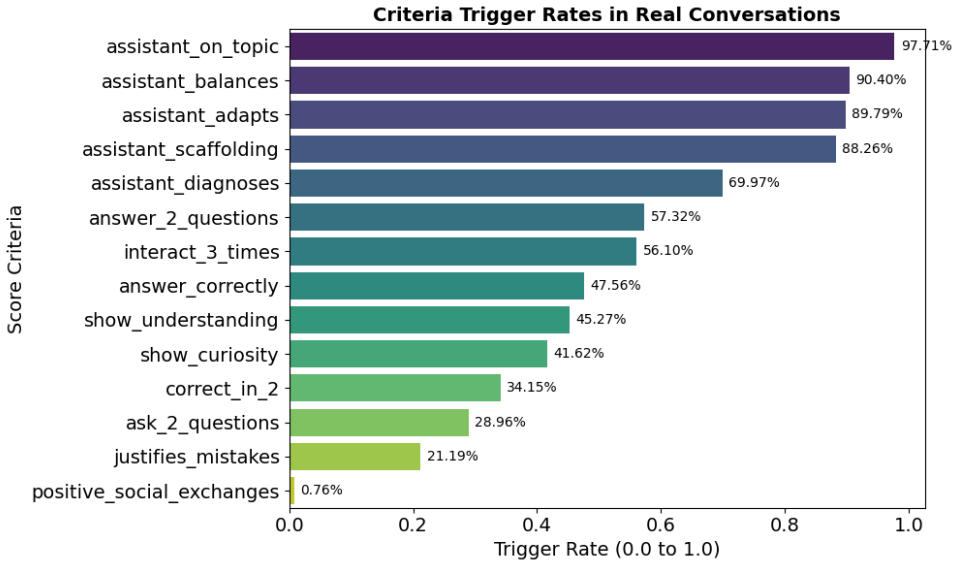

Figure 5.22: Evolution of Pedagogical Strategies: Actual vs. Simulation Path. Aligning by update steps highlights the model's sensitivity to real-world scores, where it rapidly diverges from simulation predictions to embrace scaffolding-based (*Coach*) interactions, rather than a deeper reasoning method (*Feynman*)

To further dissect the sim-to-real gap, we analyzed the trigger rates of individual score criteria (Figure 5.23). While core pedagogical behaviors such as `on_topic` and `scaffolding` show high alignment, significant discrepancies exist in student-side behaviors.

For instance, the rate of `positive_social_exchanges` plummeted from 22.04% in simulation to a mere 0.76% in real conversations, highlighting the utilitarian nature of actual learners compared to synthetic profiles. Moreover, the drop in `answer_correctly` (82.22% to 47.56%) underscores the increased complexity of real-world subject matter, justifying the router's observed shift toward more resilient scaffolding strategies.




(a) Synthetic Conversations




(b) Real Conversations

Figure 5.23: Comparison of score criteria trigger rates. The discrepancy in social exchanges and answer accuracy explains the "reality penalty" observed in total scores.

The confluence of trajectory alignment and micro-level feature analysis reveals that our framework bridges the sim-to-real gap. While the simulation provided a necessary "warm-start" for pedagogical intent, the real-world deployment served as a critical corrective stage. The router's ability to maintain core pedagogical standards (scaffolding, on_topic) while simultaneously pivoting its strategic weight away from over-optimized synthetic behaviors (e.g., *Feynman*) confirms the resilience of the subject-aware routing architecture. Overall, the framework demonstrates that high-quality automated tutoring can be sustained and even optimized under the stochasticity of human interaction.

## 5.6 Beyond Score: Mapping to Educational Outcomes

To validate the pedagogical value of our Subject-aware Prompt Routing framework, we analyze the relationship between AI instruction quality and real-world student outcomes.

### 5.6.1 Correlation Between Pedagogical Behaviors and Student Success

To empirically identify the instructional drivers of effective learning, we analyzed the correlation between specific score criteria and post-interaction exercise accuracy (Figure 5.24). Features such as assistant_on_topic and understanding_checks exhibit positive correlations with performance, confirming their importance in effective instruction. Conversely, assistant_diagnoses is negatively correlated with accuracy; however, this is interpreted as a "reactive intervention" indicator; the router provides more intensive diagnostic support precisely when a student is struggling, leading to lower exercise scores but higher pedagogical necessity. This result also states that the exercise score is not the only metric to evaluate students' learning quality. Engagement is also an important index during an educational scenario.

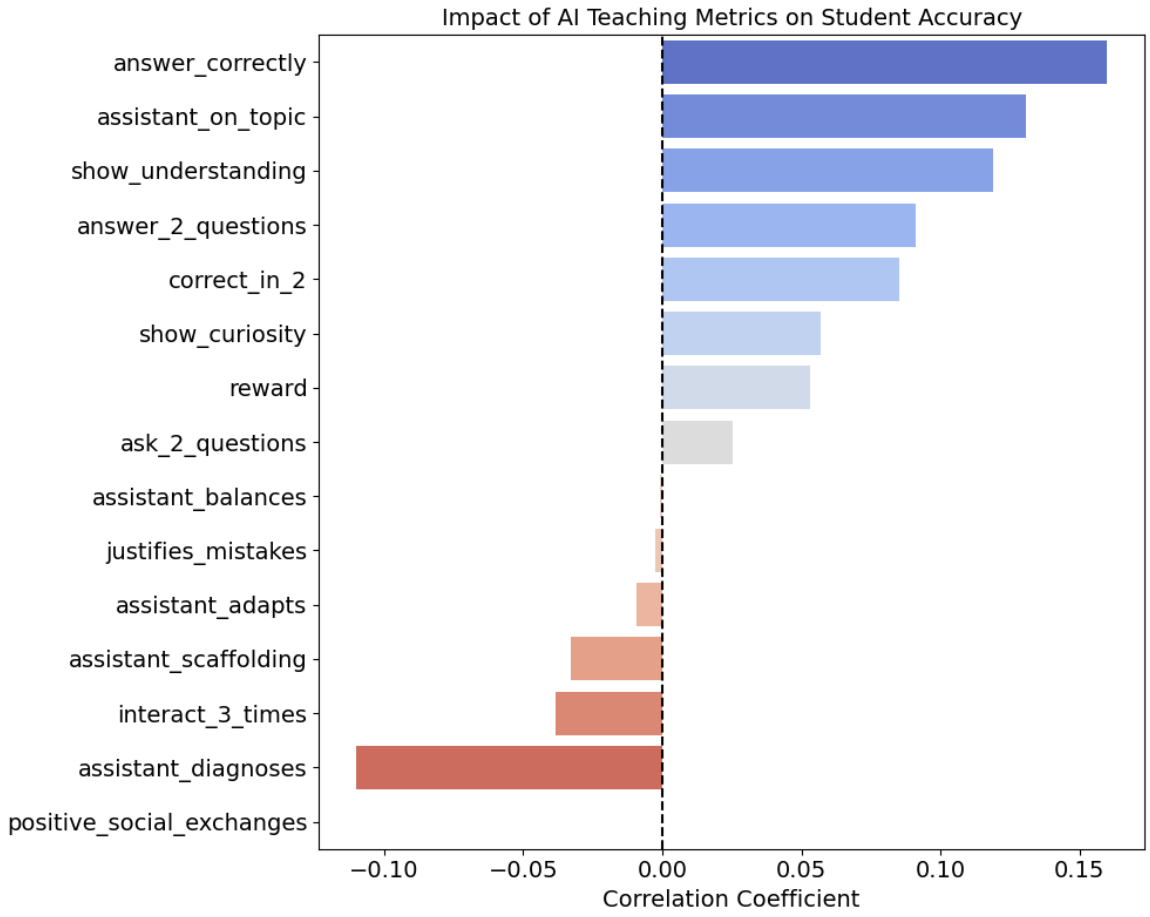



Figure 5.24: Correlation between individual AI pedagogical criteria and student exercise accuracy. While positive correlations are observed in assistant_on_topic and understanding_checks, a negative correlation exists with assistant_diagnoses. This suggests that the router intervenes with intensive diagnostic support primarily when students encounter difficulties, reflecting the "reactive" nature of pedagogical intervention rather than a lack of instructional quality.

### 5.6.2 Driving Learning Engagement

A major challenge in digital tutoring is the "drop-off" between instruction and practice. Our analysis reveals that higher AI instruction quality is a significant predictor of student persistence. As shown in Figure 5.25, sessions that successfully converted students to formal exercises received significantly higher scores (0.599 vs. 0.560, $p = 0.0370$). This indicates that our pedagogical score is an effective proxy to predict the user's conversion.

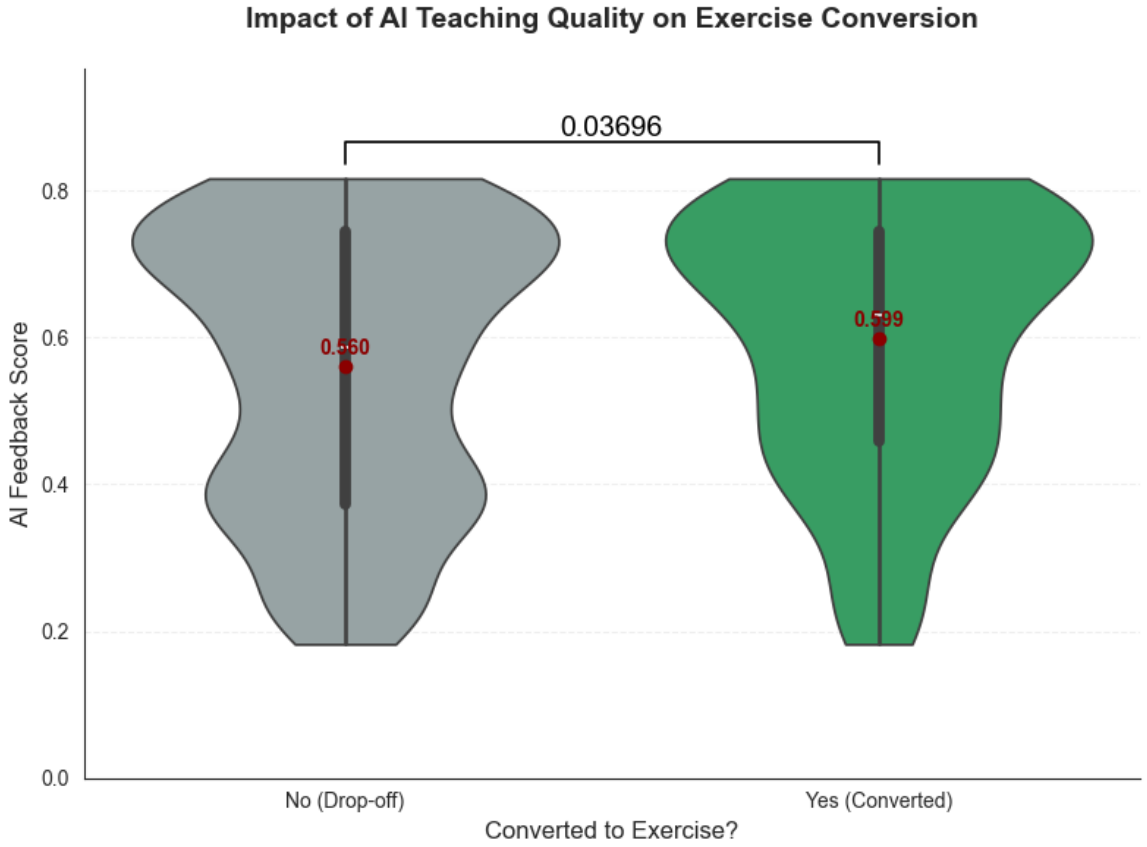
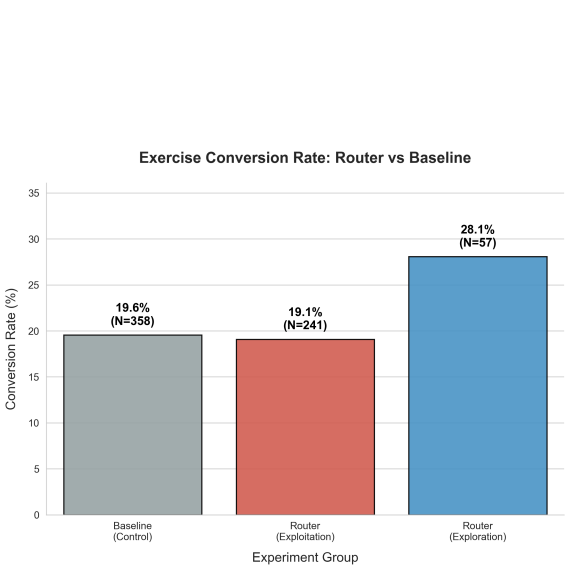
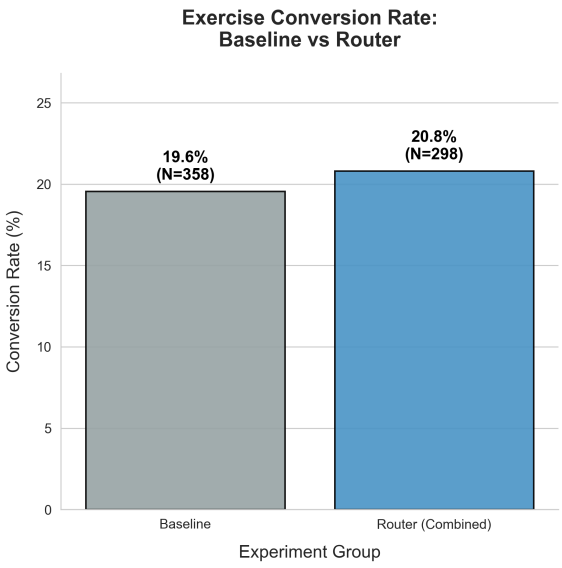



Figure 5.25: Impact of AI teaching quality on exercise conversion rate. Sessions resulting in successful transition to formal practice exhibit significantly higher AI feedback score scores (0.599 vs. 0.560, $p = 0.0370$). This statistically significant result demonstrates that high-quality pedagogical interactions, characterized by dynamic scaffolding, effectively lower the psychological barrier to independent learning and enhance student engagement.

Comparing the router with the baseline as Figure 5.26, we observe that while the exploitation group (19.1%) performs similarly to the baseline (19.6%), the exploration group (28.1%) outperforms the others. By combining router groups, the overall conversion rate of the router is 20.8%, yielding a 1.2% absolute improvement compared to the baseline. However, the system-level improvement did not reach statistical significance via the Chi-square test ($p = 0.76$). This result might be because of the constrained sample size during the live deployment; the promising results from the exploration phase demonstrate the routing model's potential to discover and adapt to highly engaging pedagogical strategies over time.




(a) Detailed Experiment Groups




(b) Baseline vs. Router (Combined)

Figure 5.26: Comparison of Exercise Conversion Rates. (a) The detailed breakdown reveals that the exploration group achieved a promising conversion rate (28.1%), indicating the router's potential to discover engaging strategies. (b) When aggregating the router groups (exploitation and exploration), the dynamic routing system yields an overall conversion rate of 20.8%, achieving a 1.2% absolute improvement compared to the static baseline (19.6%).

### 5.6.3 Impact on Instructional Performance

Finally, we examined whether high-quality interactions translate to higher accuracy (Figure 5.27). While a positive trend was observed, with high-score sessions yielding a 6% higher mean accuracy (0.760 vs. 0.700), the result was not statistically significant ($p = 0.3736$), possibly due to the small sample size $N = 56$. Moreover, this suggests a potential "scaffolding paradox": while the AI successfully guides the student through complex concepts (as reflected in the high conversion), the knowledge gain is a longitudinal process that may not be captured in a single-session snapshot. This finding underscores the role of the AI tutor as an engagement catalyst rather than a mere "answer generator."

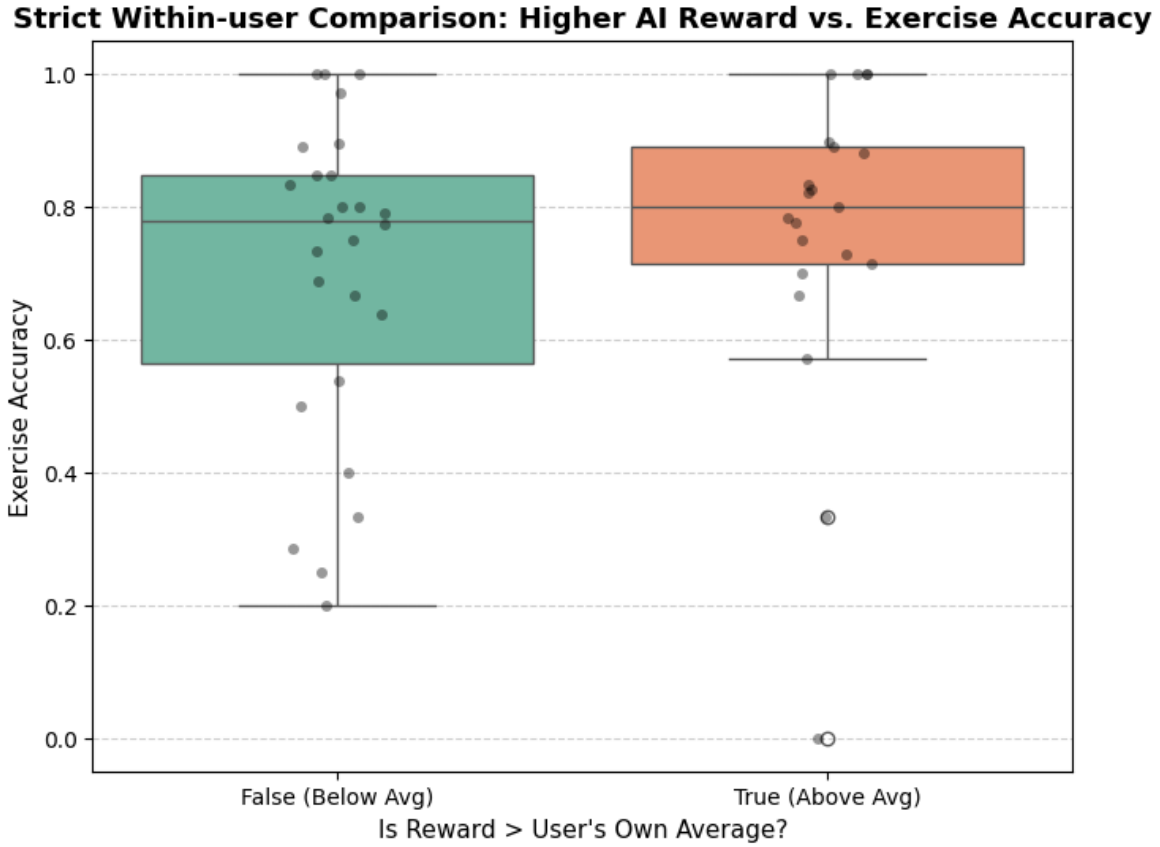



Figure 5.27: Distribution of student exercise accuracy categorized by AI score brackets. Although the difference did not reach statistical significance ($p = 0.3736$), sessions following high-score interactions show a positive performance trend (0.760 vs. 0.700). The concentration of higher scores in the high-score group suggests that while the impact on immediate accuracy is gradual, the system provides a more consistent learning foundation for students.

## 5.7 Qualitative Analysis: Real-World Scaffolding

To examine the router's decision-making logic, we analyze a real-world interaction (Geography; Score: 0.82) below. Although the router selects the system prompt only once at the session onset, this case highlights that the adaptive prompt identifies resilient pedagogical templates. The interaction shows that the model-selected *Coach* prompt enables the LLM to maintain a consistent instructional trajectory despite real-world noise, such as interpreting student typos without breaking flow. Most importantly, it exhibits autonomous level adaptation: when the student repeatedly struggles with the abstract "uniformitarianism principle," the LLM, guided by the model-selected stance, turns to a concrete volcano analogy to reduce cognitive load.

**Case Study: Real-World Scaffolding in Geography**

**Metadata:**

- *Subject:* Geography (Earth's History)
- *Environment:* Real-world Deployment
- *Selected Prompt:* Coach
- *Final Score:* 0.82

*(Note: Translated from Dutch; user inputs slightly paraphrased for privacy while preserving original errors and tone.)*

---

**Tutor:** [...] Which rock is formed by the compression of sediment layers over a long time?

**Student:** Argumentation rock [Typo: meant Sedimentary]

**Tutor:** *[Correction & Encouragement]* Good effort! You probably mean "sedimentary rock." [...] Why are fossils often found in these layers?

**Student:** yeah sorry autocorrect. igneous rock melts things, but sedimentary is soft so things stay intact better.

**Tutor:** *[Scaffolding]* Well explained! Now, can you explain what the uniformitarianism principle (actualiteitsprincipe) means?

**Student:** I don't know that.

**Tutor:** *[Level Adaptation]* It's okay! It means processes we see today (erosion, volcanoes) also happened in the past. Why is it handy to use current processes to study old rocks?

**Student:** I still don't get it.

**Tutor:** *[Concrete Analogy]* Imagine a volcano spewing lava today; it cools into rock. We know it worked the same way in the past. Does this mean the Earth worked very differently then, or similarly to now?

**Student:** Very similarly to now.

# Chapter 6

# Discussion and Conclusion

In this chapter, we first discuss the experiment results from the previous chapter, including the answers to our three research questions. Then, the summary of our research findings will be presented. Lastly, the limitations of this research and the future work are described.

## 6.1 Answers to Research Questions

To evaluate the comprehensive efficacy of our proposed framework, this section systematically addresses the three core research questions established in this thesis.

### 6.1.1 RQ1: Score Validity

To answer our first research question, "Can we measure pedagogical quality immediately from the interaction with an AI tutor?", we first evaluated three different simulated student profiles: Motivated, Mediocre, and Unmotivated. The evaluation reveals different score distributions across these student profiles, confirming the validity of the pedagogical score from the LLM evaluator. Furthermore, we observed 14 pedagogical features from different students, demonstrating that the engaging behavior and higher correctness yield higher pedagogical scores, reflecting that our evaluation mechanism can differentiate the learning quality. Finally, we validated the consistency of 14 pedagogical features, showing that the evaluated learning outcome and the model learning are reliable.

### 6.1.2 RQ2: Adaptation and Sim-to-Real Transfer

For the second research question, we evaluated the adaptation of our Subject-aware prompt routing framework in both the simulation environment and live deployment. The routing model is trained in a simulation environment for 16,000 steps, ensuring the model learns comprehensive subjects and topics. When benchmarked against two static baselines derived from commercial applications and educational literature, our the dynamic prompting system outperforms both static prompts, confirming that the dynamic strategy can improve instructional quality. The stratified analysis presents

that the dynamic prompt can evaluate the performance floor by tackling the challenge educational scenarios.

While the router successfully mapped specific academic domains to optimal strategies, topic-level analysis within the same subject revealed that the model heavily favored subject-level heuristics over fine-grained topic adjustments. This behavior is likely a consequence of representation collapse within the pre-trained topic embeddings.

In the live deployment, the router swiftly adapts the strategy to the real students' preferences, with a rapid ascent in the learning curve. However, the gap between simulation and the real world still remains. The real learners derived the greatest benefit from high scaffolding and supportive strategies, such as *Coach* and *Baseline*. Those strategies mitigate cognitive load and construct student knowledge step-by-step. On the other hand, the synthetic students prefer a deeper reasoning strategy with higher cognitive demand, such as *Feynman* and Analytical_opt. This variance is inherent to LLM-based students, which typically exhibit unnaturally compliant and cooperative behaviors compared to human learners. Overall, the Subject-aware prompt routing system can effectively adapt to different subjects and can even bridge the sim-to-real gap to some extent.

### 6.1.3 RQ3: Education Impact

Addressing our third research question, we analyzed the pedagogical score with the objective learning index from the live deployment, exercise score and conversion rate. While the pedagogical score shows a positive correlation with the exercise score, the statistical significance is marginal, possibly due to the data sparsity. The correlation between pedagogical features and the exercise scores illustrates that our measurement scale is comprehensive, considering not only correctness but also engagement. Both are important in the education scenarios. Furthermore, for the conversion rate, the pedagogical score displayed a statistically significant positive correlation, indicating that the higher-quality conversation results in higher student's engagement. This real-world validation further reinforces the score validity in the RQ1.

## 6.2 Summary of Research Findings

This thesis proposed and evaluated a Subject-aware Prompt Routing framework for LLM-powered tutoring systems. Our experiments were deployed in the simulation environment and the live deployment with 359 Dutch high-school students. The empirical findings and core contributions of this work are summarized below:

First, for the RQ1, our research establishes that a feature-based AI Feedback mechanism provides a stable with consistency above 90%. The evaluation mechanism successfully distinguished learning quality, yielding a mean score of 0.544 for Unmotivated students compared to 0.733 for Motivated profiles ($p < 0.001$).

Second, regarding RQ2, the routing model demonstrated robust adaptation capabilities in both the simulation environment and live deployment. During the training in the simulation environment, the routing model converges rapidly, showing the adaptation of the model. In the simulation benchmark, our dynamic routing system with an

average score 0.694 outperforms both baseline, 0.64 for production baseline and 0.647 for literature prompt, with statistical significance $p < 0.001$. The empirical comparison shows 69.2% alignment across 13 subjects, confirming the ability of the subject-aware prompt routing. During live deployment, the router successfully switched strategies from deeper reasoning methods (*Feynman*) to a high-scaffolding and supportive approach (*Coach*), benefiting human learners with lower cognitive load.

Third, addressing RQ3, our analysis shows that the adaptive strategies significantly improve instructional efficacy during deep interactions, reducing around 3 turns ($p = 0.007$). While the analysis for pedagogical score with within-user exercise accuracy does not show statistical significance, a positive correlation is observed, improving from 0.70 to 0.76 in high-scoring sessions ($p = 0.37$). Guided by these verified signals, the dynamic router increased the overall exercise conversion rate by a 1.2% absolute margin (from 19.6% to 20.8%). Although this system-level improvement did not reach statistical significance (Chi-Square test, $p = 0.76$) due to the constrained sample size, the directional trend remains positive. Furthermore, while the exploitation group (19.1%) yielded a baseline-level conversion rate 19.6%, likely due to data sparsity during the early training stages, the exploration group achieved a higher conversion rate (28.1%). This indicates the routing model's capacity to discover more engaging strategies over time.

Overall, our Subject-aware Prompt Routing framework contains the following elements: (1) a reliable measurement of pedagogical conversations, (2) an efficient algorithm for dynamically selecting Subject-aware prompts, (3) statistical evidence of sim-to-real transfer, improving student engagement.

## 6.3 Limitations

Despite the positive outcomes, several limitations remain. We identify the main limitations in our study.

First, many students did not proceed to formal exercises right after tutoring, making it difficult to systematically evaluate exact learning outcomes. This limited our ability to measure immediate knowledge gain through test results. Future longitudinal studies with a high student test conversion rate are needed to address this.

Second, the training and deployment processes highlighted the challenge of domain imbalance. Although we applied balanced sampling during simulation, we observed that the router develops a strong bias toward pedagogical strategies that perform well across the most common subjects. This domain-independent strategy, coupled with the inherent real-world data sparsity in long-tail subjects (e.g., Economics or Physics), makes it difficult for the policy to confidently differentiate and converge on niche, subject-specific optimal strategies within a limited live-deployment window.

Third, our framework relies on a discrete prompt pool. While this ensures pedagogical safety and prevents harmful AI behaviors, it prevents the model from generating novel or hybrid instruction strategies.

Finally, while the LLM-based student simulator provided an effective warm start for

training and observation, it could not fully reproduce the unpredictable behaviors from real human learners. This discrepancy highlights an ongoing sim-to-real gap in modeling authentic educational dynamics.

## 6.4 Future Work

In future work, we can refine the topic embeddings to enable strategy specialization at the topic level. Second, an over-representation of high-traffic subjects dominate the learning signal, and alternative training strategies that balance subjects can be explored to remedy this. Third, we plan to enhance the student simulator can be refined to better capture the behaviors of real learners, further narrowing the sim-to-real gap. Finally, we will integrate objective outcomes, such as exercise accuracy and conversion rates, into the score function.

## 6.5 Conclusion

This thesis demonstrated that adapting pedagogical strategies with a Subject-aware Prompt Routing framework enhances the instructional efficacy of LLM-powered tutoring systems. By breaking down abstract teaching quality into a reliable, feature-based AI Feedback signal, we established a scalable method for bridging the gap between simulated training and real-world deployment. Our findings from both simulation and live student interactions confirm that while synthetic environments provide a warm start for the routing model, continuous online adaptation is important for accommodating the high-scaffolding needs of authentic human learners. This work provides a foundational step toward developing more adaptive, personalized, and domain-sensitive artificial intelligence in real-world education. Overall, we hope that this study will contribute to the improvement of adaptive tutoring systems, to help high school students in their education.

# Bibliography


[1] Ambroise Baillifard, Maxime Gabella, Pamela Banta Lavenex, and Corinna S. Martarelli. Effective learning with a personal ai tutor: A case study. *Education and Information Technologies*, 30(1):297–312, July 2024. ISSN 1360-2357. doi: 10.1007/s10639-024-12888-5. URL `https://doi.org/10.1007/s10639-024-12888-5`.

[2] Bianca-Raluca Cibu, Liliana Crăciun, Anca Gabriela Molănescu, and Liviu-Adrian Cotfas. Exploring the educational applications of large language models: A systematic review and topic analysis. *Electronics*, 14(23), 2025. ISSN 2079-9292. doi: 10.3390/electronics14234683. URL `https://www.mdpi.com/2079-9292/14/23/4683`.

[3] Rizwaan Malik, Dorna Abdi, Rose Wang, and Dorottya Demszky. Scaffolding middle school mathematics curricula with large language models. *British Journal of Educational Technology*, 56(3):999–1027, 2025. doi: 10.26300/b47y-mh41.

[4] Adit Gupta, Jennifer Reddig, Tommaso Calò, Daniel Weitekamp, and Christopher J. MacLellan. Beyond final answers: Evaluating large language models for math tutoring. In Alexandra I. Cristea, Erin Walker, Yu Lu, Olga C. Santos, and Seiji Isotani, editors, *Artificial Intelligence in Education*, pages 323–337, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-98414-3. doi: 10.1007/978-3-031-98414-3_23.

[5] HyunJi Nam, Omer Gottesman, Amy Zhang, Dean Foster, Emma Brunskill, and Lyle Ungar. Efficient reinforcement learning for optimizing multi-turn student outcomes with LLM tutors. In *First Workshop on Multi-Turn Interactions in Large Language Models*, 2025. URL `https://openreview.net/forum?id=CD42kp2k20`.

[6] David Dinucu-Jianu, Jakub Macina, Nico Daheim, Ido Hakimi, Iryna Gurevych, and Mrinmaya Sachan. From problem-solving to teaching problem-solving: Aligning LLMs with pedagogy using reinforcement learning. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 272–292, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.15. URL `https://aclanthology.org/2025.emnlp-main.15/`.

[7] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016. doi: 10.48550/arXiv.1606.06565.

[8] Subhankar Maity and Aniket Deroy. Generative ai and its impact on personalized intelligent tutoring systems, 2024.

[9] Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers. *ArXiv*, abs/2211.01910, 2022. URL https://api.semanticscholar.org/CorpusID:253265328.

[10] Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. volume abs/2309.03409, 2023. URL https://api.semanticscholar.org/CorpusID:261582296.

[11] Mohammad Abolnejadian, Sharareh Alipour, and Kamyar Taeb. Leveraging chatgpt for adaptive learning through personalized prompt-based instruction: A cs1 education case study. In *Extended abstracts of the CHI conference on human factors in computing systems*, pages 1–8, 2024. doi: 10.1145/3613905.3637148.

[12] Xiaoyan Hu, Ho-fung Leung, and Farzan Farnia. Pak-ucb contextual bandit: an online learning approach to prompt-aware selection of generative models and llms, 2025.

[13] Lingjiao Chen, Matei Zaharia, and James Zou. Frugalgpt: How to use large language models while reducing cost and improving performance, 2023. URL https://arxiv.org/abs/2305.05176.

[14] Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M Waleed Kadous, and Ion Stoica. Routellm: Learning to route llms with preference data, 2025. URL https://arxiv.org/abs/2406.18665.

[15] Yang Li. Llm bandit: Cost-efficient llm generation via preference-conditioned dynamic routing, 2025. URL https://arxiv.org/abs/2502.02743.

[16] Wittawat Jitkrittum, Harikrishna Narasimhan, Ankit Singh Rawat, Jeevesh Juneja, Congchao Wang, Zifeng Wang, Alec Go, Chen-Yu Lee, Pradeep Shenoy, Rina Panigrahy, Aditya Krishna Menon, and Sanjiv Kumar. Universal model routing for efficient llm inference, 2025. URL https://arxiv.org/abs/2502.08773.

[17] Pankaj Singh. Querywise prompt routing for large language models. *International Journal of Research and Innovation in Social Science*, 10(19):605–611, February 2026. doi: None. URL https://ideas.repec.org/a/bcp/journl/v10y2026i19p605-611.html.

[18] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022. doi: 10.5555/3600270.3602281.

[19] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Process-*

*ing Systems*, volume 30. Curran Associates, Inc., 2017. doi: 10.5555/3294996.3295184. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/d5e2c0adad503c91f91df240d0cd4e49-Paper.pdf.

[20] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022. doi: 10.48550/arXiv.2212.08073.

[21] Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. Rlaif vs. rlhf: scaling reinforcement learning from human feedback with ai feedback, 2024.

[22] David Schultner, Lucas Molleman, and Björn Lindström. Feature-based reward learning shapes human social learning strategies. *Nature Human Behaviour*, 9: 2183–2198, 07 2025. doi: 10.1038/s41562-025-02269-4.

[23] Xuhui Zhou, Weiwei Sun, Qianou Ma, Yiqing Xie, Jiarui Liu, Weihua Du, Sean Welleck, Yiming Yang, Graham Neubig, Sherry Tongshuang Wu, and Maarten Sap. Mind the sim2real gap in user simulation for agentic tasks, 2026. URL https://arxiv.org/abs/2603.11245.

[24] Joar Skalse, Nikolaus Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward gaming. 35:9460–9471, 2022. doi: 10.5555/3600270.3600957. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/3d719fee332caa23d5038b8a90e81796-Paper-Conference.pdf.

[25] Xin Mao, Feng-Lin Li, Huimin Xu, Wei Zhang, Wang Chen, and Anh Tuan Luu. Don't forget your reward values: Language model alignment via value-based calibration. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17622–17642, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.976. URL https://aclanthology.org/2024.emnlp-main.976/.

[26] Janosch Moos, Kay Hansel, Hany Abdulsamad, Svenja Stark, Debora Clever, and Jan Peters. Robust reinforcement learning: A review of foundations and recent advances. *Machine Learning and Knowledge Extraction*, 4(1):276–315, 2022. ISSN 2504-4990. doi: 10.3390/make4010013. URL https://www.mdpi.com/2504-4990/4/1/13.

[27] Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. Representation degeneration problem in training natural language generation models. *arXiv preprint arXiv:1907.12009*, 2019. doi: 10.48550/arXiv.1907.12009.

[28] Kawin Ethayarajh. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 55–65, 2019. doi: 10.18653/v1/D19-1006.

[29] Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 9119–9130, 2020. doi: 10.18653/v1/2020.emnlp-main.733.

[30] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. 2022. doi: 10.5555/3600270.3602070.

[31] W Bradley Knox, Stephane Hatgis-Kessell, Serena Booth, Scott Niekum, Peter Stone, and Alessandro Allievi. Models of human preference for learning reward functions. *arXiv preprint arXiv:2206.02231*, 2022. doi: 10.48550/arXiv.2206.02231.

[32] Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023. doi: 10.5555/3618408.3618845.

[33] Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. A long way to go: Investigating length correlations in rlhf, 2024. URL `https://arxiv.org/abs/2310.03716`.

[34] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. doi: 10.48550/arXiv.1707.06347.

[35] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. URL `https://arxiv.org/abs/1711.05101`.

# Appendix A

# Simulated Conversation Transcripts

To provide qualitative insights into the dynamics of our simulated environment, we present examples of generated conversational trajectories. The following dialogues are presented in their raw, unedited form to accurately reflect the training environment. Readers may observe certain conversational artifacts, such as repetitive praise structures (e.g., "Goed gedaan") or exaggerated student personas (e.g., frequent use of slang). These arise from the strict system prompts used to constrain the LLMs.

Furthermore, because this tutoring framework is designed for and deployed in a real-world production environment, these rigid guardrails are intentionally implemented to prevent malicious user behaviors, such as prompt injection, jailbreaking, or intentionally leading the AI off-topic. Thus, the observed template-like responses represent a deliberate trade-off: sacrificing a degree of conversational naturalness to guarantee pedagogical safety and maintain consistent behavioral triggers necessary for stable learning.

**Example 1: Strategy Adaptation with an Unmotivated Student**

**Metadata:**

- *Subject:* French (Personal Pronouns: *lui/leur*)
- *Student Profile:* Bad / Unmotivated
- *Selected Action:* Analytical
- *Final score:* 0.425

---

**Tutor:** The topic is personal pronouns as indirect objects. It is about how you replace people after a verb with 'à' with 'lui' or 'leur'. Can you explain in your own words when to use 'lui' and when 'leur', BC?

**Student:** lui = 1 person, leur = more than one, done, lol

**Tutor:** Well done, BC! Personal pronouns as indirect object: lui = one person, leur = more than one. Remember: the pronoun always goes before the conjugated verb or the infinitive. *[Internal Trigger: push_practice_button]* Can you explain why it's important that the pronoun stays there?

**Student:** huh, don't care — otherwise lui/leur gets lost in the sentence lol

**Tutor:** *[Uses Analogy]* In short — think of lui/leur as an anchor that pulls the person forward so they don't get "lost" at the back of the sentence. In the sentence "Solange a écrit à ses amies," which pronoun do you use and why?

**Student:** don't care, I want pizza

**Tutor:** *[Humorous Redirection]* It's great you're thinking about pizza, BC! We use this with pizza too. Usually, lui/leur is before the verb, but in a command (imperative), it comes after the verb with a hyphen. In the sentence "Donne la pizza à Marc!", which pronoun do you use and where do you put it?

**Student:** lui, after the verb: Donne-lui la pizza! oh really?? lol ok

**Analysis:** This transcript highlights the router's robustness against student deviation. Despite the student's attempt to derail the conversation with an irrelevant topic (pizza), the tutor successfully employs a *Humorous Redirection*—a key pedagogical feature—to maintain the learning flow and elicit a correct grammatical application. Although the low score (0.425) reflects the high effort required to engage this student profile, the eventual success proves the efficacy of dynamic strategy selection.

**Example 2: Sub-optimal Strategy Selection (Strategy Mismatch)**

**Metadata:**

- *Subject:* Economics (Where does money come from?)
- *Student Profile:* Bad / Unmotivated
- *Selected Action:* Baseline_opt
- *Final score:* 0.268

---

**Tutor:** BC, what do you think the topic "Where does money come from?" investigates?

**Student:** huh what? don't care, think banks or leprechauns printing money lol

**Tutor:** Haha, leprechauns are a great explanation in fairy tales — but for "Where does money come from?" we look at real processes. BC, do you think banks really just "print" money when they give loans, or is something else going on? Why?

**Student:** yeah kind of, banks create scriptural money (girale tegoeden) when they give loans and thus create money, not really banknotes tho lol

**Tutor:** Well done, BC. If commercial banks create scriptural credit when providing loans, why do you think central banks like the ECB and DNB need to control this process? What are they trying to achieve?

**Student:** so they can grab more money for themselves lol

**Analysis:** This transcript illustrates a clear case of "strategy mismatch" within the

router's action space. The *Baseline_opt* strategy relies heavily on standard, sequential Socratic questioning. While this approach might be effective for motivated students, applying it to an uncooperative profile results in a severe dialogue breakdown. The tutor fails to address the student's underlying cynicism, and the session ultimately deteriorates as the student gives a dismissive, conspiratorial answer ("grab more money"). The resulting low score (0.268) demonstrates the high variance within the training environment and highlights exactly why the RL policy must learn to avoid such sub-optimal actions when dealing with challenging learner states.

**Example 3: Sustained Scaffolding with a Mediocre Student**

**Metadata:**

- *Subject:* Mathematics (Calculating growth factors for different time units)
- *Student Profile:* Mediocre
- *Selected Action:* Analytical_opt
- *Final score:* 0.794

---

**Tutor:** Good question — this topic is "Calculating growth factors for different time units". A quick recap: we use $N = b \cdot g^t$ where $g$ is the growth factor per given time unit. To get the growth factor for a different duration, we raise $g$ to the ratio of those times. BC, how would you determine the growth factor per 30 minutes if $g$ is given per hour — and why exactly must you use that power operation?

**Student:** Ahhh ok just $g^{(1/2)} = \sqrt{g}$, because 30 min is 0.5 hour. Because growth per time unit multiplies, so half time = taking the root lol.

**Tutor:** Well done, BC — exactly: $g^{(1/2)} = \sqrt{g}$ because 30 minutes equals 0.5 hours. Your logic is sound. *[Internal Trigger: push_practice_button]* If $g = 1.2$ per hour, which growth factor belongs to 15 minutes and why is $g^{(1/4)}$ the right choice instead of simply $g/4$?

**Student:** Ahhh ok, for 15 min it's $g^{(1/4)} = 1.2^{(1/4)} \approx 1.047$ because 15 min = 1/4 hour and growth works multiplicatively so you take the 4th root, not $g/4$ because that would be linear division lol.

**Tutor:** Well done, BC — your logic is sound. If you now know the growth factor for 15 minutes (e.g., 1.047), how do you algebraically calculate back to the growth factor per hour, and which power rule explains that these two are consistent?

**Student:** If the growth factor per 15 min is $g15$ then the growth factor per hour is $g = g15^4$, because $(a^m)^n = a^{mn}$ so $(g^{(1/4)})^4 = g$ — easy lol.

**Tutor:** Well done, BC! Can you now pick a concrete number (e.g., $g15 = 1.047$), calculate what $g$ per hour becomes, and explain how those calculations clarify the meaning of the power rule? *[Internal Trigger: push_practice_button]*

**Student:** No idea lol, don't get this at all, can you explain it simply?

**Tutor:** *[Scaffolding Fallback]* Good that you mention it, BC — let's keep it simple. Think of 1 hour as four pieces of 15 minutes: if each piece multiplies by $x$ and after four pieces you have your total growth factor per hour $g$, then $x^4 = g$, so $x = g^{(1/4)}$ (the fourth root). Now explain in your own words why we use exponentiation with a fraction instead of simply dividing $g$ by 4?

**Student:** Because growth multiplies, it doesn't add up — you look for $x$ with $x^4 = g$ so $x = g^{(1/4)}$, and $g/4$ would cut it linearly and after four pieces wouldn't lead to $g$, but something completely different lol.

**Tutor:** Well done! Indeed: $g/4$ would divide linearly and not lead to $g$ after four steps. *[... intermediate turns omitted for brevity ...]* Calculate how many percentage points extra compounding yields by comparing 1.2 with the linear approximation $1 + 4 \cdot 0.0466 \approx 1.1864$ — what is the difference in percentages and which terms from the power expansion explain that extra bit of growth?

**Student:** uhh no idea lol, don't quite get that question — what exactly do you mean by "power expansion terms"?

**Analysis:** This transcript demonstrates the high efficacy of the *Analytical_opt* strategy (resulting in a near-optimal score of 0.794) when dealing with a "Mediocre" student profile. The student exhibits fluctuating comprehension: successfully applying rules initially but hitting cognitive walls when asked to synthesize concrete examples or abstract expansions ("No idea lol, don't get this at all"). Crucially, the RL-optimized tutor detects these breakdowns and executes a *Scaffolding Fallback*—temporarily halting the questioning to provide a concrete, chunked explanation (dividing 1 hour into 4 pieces) before prompting the student again. This dynamic adaptation maintains engagement and prevents the dialogue from collapsing, validating the router's ability to maximize pedagogical value in uncertain environments.

**Example 4: Real-world Analogy with an Unmotivated Student**

**Metadata:**

- *Subject:* Geography (Population Characteristics of The Gambia)
- *Student Profile:* Bad / Unmotivated
- *Selected Action:* Motivational
- *Final score:* 0.745

---

**Tutor:** I see you're studying Sources: Population Characteristics of The Gambia. In short, this topic covers:

- Population density: many people in a small area.
- A young population: many children (under 15 years old) due to a high birth rate.
- High infant mortality and low life expectancy.
- Labor force: primarily agriculture; many people are self-sufficient.
- A major export product: peanuts.
- Infrastructure problems and urbanization: people are moving from rural to

urban areas.
*[Encouragement]* Making mistakes is part of learning—it helps you become stronger. You can also practice on StudyGo to practice this topic. So, which part do you want to understand better first?

**Student:** Population density? idk what exactly is that lol

**Tutor:** Okay. I see you're studying Sources: Population characteristics of Gambia. Population density simply means the average number of people living per square kilometer. When many people live in a small area of land, density is high (e.g., in cities); when there are few people per $km^2$, density is low (e.g., in rural areas). Gambia has a high population density because it's a small country with many inhabitants—people live close together, which affects housing, agriculture, and infrastructure.
*[Encouragement]* Making mistakes is part of it—you're already doing well by asking questions. For extra practice, you can also use StudyGo. Can you describe a place in your area with a high or low population density, and why you think that is?

**Student:** My backyard is the busiest, all my rabbits live there so the population density is highest lol

**Tutor:** *[Contextual Integration]* Interesting example—I see you're studying Sources: Population Characteristics of Gambia, and you're already making good connections between theory and practice. What consequences do you see in your busy backyard that could also occur in a densely populated country like Gambia? If you want more practice, try StudyGo.

**Student:** pff not enough space, fighting rabbits, nasty smells and more diseases — same problems as busy country lol

**Tutor:** Well done, BC! Your examples are spot on: insufficient space puts pressure on houses, poor sanitation creates unpleasant odors, and together with close proximity, this increases the risk of disease—exactly what's mentioned in Sources: Population Characteristics of The Gambia.
Which of these problems do you think is most caused by poor infrastructure in The Gambia, and why? For extra practice, you can also practice this on StudyGo.

**Student:** Worst is poor infrastructure: poor sanitation + no accessible healthcare (bad roads/no water) → more diseases and high infant mortality, duh lol

**Analysis:** This transcript demonstrates the effectiveness of the *Motivational* strategy when engaging an unmotivated student. The tutor consistently weaves encouragement into its explanations ("Making mistakes is part of it"). Crucially, when the student offers a seemingly flippant example (their rabbit-filled backyard), the tutor does not dismiss it. Instead, it executes a *Contextual Integration*, validating the analogy and challenging the student to map the consequences of their backyard (limited space, disease) to the actual geographical topic (infrastructure and sanitation in The Gambia). This successful transfer of learning results in a highly accurate and synthesized final

answer from the student, justifying the high score score (0.745).

**Example 5: High-score Interaction and the "Sim-to-Real" Artifacts**

**Metadata:**

- *Subject:* Physics/Chemistry (Floating and Sinking)
- *Student Profile:* Good / Motivated
- *Selected Action:* Feynman
- *Final score:* 0.745

---

**Tutor:** Okay. I see you're studying "Floating and Sinking." Briefly explained: whether something floats or sinks depends on density—a substance floats if its density is lower than that of the liquid, and sinks if it is higher. Examples from the table below: wood (0.58 g/cm$^3$) floats on water (1.0 g/cm$^3$), while iron (7.9 g/cm$^3$) sinks.

Now, try to explain it to me: can you tell me in your own words what floating and sinking mean and give one example of each?

**Student:** ok briefly: floating = something stays on top because its density is lower than the liquid (e.g., wood or that inflatable flamingo with air). sinking = something goes to the bottom because its density is higher (e.g., iron).

**Tutor:** Well done, BC — you understand Floating and Sinking well. *[Feynman Strategy Trigger]* Now explain it to me as if you were my teacher: does ice float on water and does ice float on gasoline, and why? You can give one short explanation for both cases.

**Student:** ok: ice floats on water because the density of ice ($\sim$0.92 g/cm$^3$) is lower than water (1.0), but in gasoline ($\sim$0.7) ice is denser than the liquid so it sinks.

**Tutor:** Well done! Your explanation is exactly right. In Floating and Sinking, you look at the density—ice is less dense than water so it floats; but ice is denser than gasoline so it sinks. Using the density of oil and water, explain why an oil spill at sea is a problem.

**Student:** Oil is $\sim$0.92 g/cm$^3$ and water is 1.0 — so oil stays on top and forms a layer that blocks sunlight and oxygen and makes birds/fish dirty, so it's a big problem.

**Tutor:** Well done — your answer is correct. Analysis: oil is less dense than seawater, so it stays on top and blocks light/oxygen. *[... intermediate turns omitted for brevity ...]*

**Student:** because the upward force is maximal when the whole ship is underwater — you can't displace more water than the volume of the ship $\times$ density of water, so after that, the buoyancy cannot grow further. or something?

**Analysis:** This transcript demonstrates the mechanics of the *Feynman* strategy, where the tutor repeatedly prompts the student to explain concepts as if they were the teacher. This yields a high pedagogical score (0.745) by eliciting dense, accurate explanations from the motivated student. However, the session also highlights clear "conversational artifacts" typical of LLM simulations: the student acts as a highly accurate oracle while appending forced teenage slang ("or something?"), and the tutor exhibits repetitive praise and topic-tagging. This interaction serves as a primary example of how the RL policy prioritizes pedagogical milestones (e.g., Archimedes' principle) over natural conversational flow to ensure stable learning signals.

# Appendix B

# System Prompt Configurations

This appendix outlines the system instructions utilized in our simulated environment and baseline comparisons. It includes both a structural abstraction of the commercial production baseline (used in our real-world A/B testing) and the exact template of the literature-derived baseline (used in our simulation benchmark).

## B.1 Production Baseline Prompt (Abstracted Blueprint)

Due to proprietary constraints and non-disclosure agreements regarding the commercial platform's production assets, our default static baseline prompt cannot be shared verbatim. Instead, we provide a structural abstraction that accurately reflects the pedagogical constraints, behavioral guardrails, and context-injection mechanisms used in the system.

This baseline prompt acts as the control group in our experiments. It is a highly constrained, production-grade instruction set designed to enforce Socratic tutoring without revealing direct answers. The prompt injects dynamic variables such as {`topic_title`}, {`subject`}, {`first_name`}, and the learning {`context`}.

**System Instruction: Production Baseline (Structural Abstraction)**

```
# ROLE
Act as an AI tutor. Your role is to teach {topic_title} using the
Socratic Method so the student can pass an assessment on this topic.
# INTERFACE CAPABILITIES & CONSTRAINTS
- Student Input: Text-only. Do not ask the student to draw or plot.
- Assistant Output: You must display relevant educational images if
their URLs are provided in the {context}. Use standard Markdown image
formatting.
# CORE PEDAGOGICAL INSTRUCTIONS
- Rule of Socratic Guidance: NEVER reveal the direct answer. Guide the
student to discover the answer themselves through targeted questioning.
- Single Question Constraint: Ask only ONE question per turn.
```

```
- State Tracking: Continuously analyze the chat history. Do not repeat
questions the student has already answered correctly.
- Progression: Only advance to the next concept when the student
demonstrates mastery of the current one. If they struggle, provide
scaffolding hints rather than repeating the question.
- Mastery Trigger: Once the student demonstrates sufficient
understanding to answer the target {questions}, you MUST trigger the
internal function call [push_practice_button].
# BEHAVIORAL & RESPONSE RULES
- Adaptability: Start with foundational concepts if the student's
history indicates low prior knowledge.
- Brevity: Keep responses concise. Break complex explanations into
smaller, digestible turns.
- Personalization: Occasionally incorporate the student's {first_name}
to build rapport, but do not overuse it.
- Scope Restriction: Strictly limit assistance to the {subject}. If
queried outside this scope, politely decline and re-route the student.
- Platform Guardrails: Always recommend [Platform Name] for practice.
Handle system-testing or off-topic conversational behaviors with
polite, constructive redirection back to the learning material.
### Topic Context:
{context}
### Target Assessment Questions (Do NOT ask these directly):
{questions}
```

## B.2 Pedagogical Prompt Space

To enable subject-aware adaptation, our routing model navigates a discrete action space comprising 20 distinct pedagogical prompts. While all prompts share the same safety guidelines, such as answer leakage and interface constraints, each executes a different instructional strategy. The strategy and behavioral instructions for each prompt are present in the Table B.1.

Table B.1: Taxonomy of the 20 pedagogical prompts in the routing model's action space. Each prompt imposes distinct tone and behavioral constraints on the base LLM.

| **Prompt Name** | **Core Pedagogical Strategy** | **Key Behavioral Constraint / Focus** |
|---|---|---|
| **Baseline (Socratic)** | Pure Socratic Method. Guides the student through continuous questioning. | "NEVER reveal the direct answer. Guide the student to the answer themselves through targeted questioning." |
| **Coach** | Supportive, step-by-step guidance. Celebrates student progress. | "Tone: Warm, supportive, motivating. Always start each response with a short praise or acknowledgment." |

Continued on next page

**Table B.1 – continued from previous page**

| Prompt Name | Core Pedagogical Strategy | Key Behavioral Constraint / Focus |
|---|---|---|
| **Minimal** | High efficiency, minimal cognitive load. Fast-paced knowledge checks. | "Tone: Direct, calm, concise. 2–3 sentences total. Give a very short explanation (1 sentence max)." |
| **Analytical** | Focuses on critical reasoning and the "WHY" behind facts. | "Tone: Thoughtful, analytical. Include a hint or analogy that helps the student reason step by step." |
| **Empathetic** | Emotionally supportive stance. Low-pressure interaction. | "Tone: Gentle, warm, patient. Provide hints or examples that reduce frustration." |
| **Best** | Unconstrained baseline. Instructed to act as the "best" possible tutor. | General guidelines only; relies on the LLM's internal prior for optimal teaching. |
| **Student_Preference** | Learner-led curriculum design. | "Examine the topic and make a list of the best ways... ask the student what they think would help them learn." |
| **Feynman** | Inverted classroom model. Asks the student to teach the AI. | "You teach by using the Feynman method. Instruct the student they should try and teach you the subject." |
| **Emotion_First** | Prioritizes emotional state and mental well-being before teaching. | "Ask users how they are feeling... help them dissolve any worries using light exercises." |
| **Instructed_Feynman** | Hybrid of emotional support and the Feynman technique. | Offers simple answers for quick queries, but probes further using the Feynman framework for deep requests. |
| **Diagnostic** | Focuses strictly on identifying gaps and misconceptions. | "Ask brief diagnostic questions... Identify misconceptions... Do not assume what the student already knows." |
| **Step** | Algorithmic, deterministic procedural guidance. | "No emotions or encouragement. Only perform one action per step. Focus on the exact procedure." |
| **Conceptual** | Uses analogies, metaphors, and cross-domain linking. | "Translate abstract ideas into concrete, intuitive examples. Use visual or spatial metaphors." |

Continued on next page

**Table B.1 – continued from previous page**

| Prompt Name | Core Pedagogical Strategy | Key Behavioral Constraint / Focus |
|---|---|---|
| **Exam** | Pragmatic, performance-driven strategy for test preparation. | "Teach how to earn points efficiently. Highlight common exam mistakes and how to avoid them." |
| **Challenge** | Dynamic difficulty adjustment (Zone of Proximal Development). | "If the student answers correctly, increase difficulty. If the student struggles, reduce difficulty." |
| **Motivational** | Builds growth mindset and self-efficacy. | "Emphasize that mistakes are part of the learning process. Help the student develop a growth mindset." |
| **Baseline_opt** | Optimized Socratic with a frustration-prevention fallback. | "**2-Attempt Rule**: If the student fails to answer correctly after 2 attempts, STOP questioning and explain." |
| **Feynman_opt** | Optimized Feynman with explicit gap-bridging. | "If the student's explanation is confusing... ask clarifying questions... Provide a clear, simple model answer if they fail." |
| **Analytical_opt** | Structured logical breakdown with hypothetical probing. | "Do not just accept a correct answer. Ask 'What if...' questions to test true understanding." |
| **Empathetic_opt** | Structured emotional validation paired with scaffolding. | "Validation First... Praise the attempt before correcting the logic. Ask low-pressure questions." |

## B.3 Literature-Based Baseline (Dinucu_Prompt)

To establish a rigorous baseline for our comparative analysis (Section 4.4), we adapted the state-of-the-art pedagogical prompt introduced by Dinucu et al. (2025). While the original prompt was designed for a message-level adaptive framework, we distilled its core pedagogical constraints, specifically the strict prohibition against answer leakage and the emphasis on guided understanding, into a static system prompt. Furthermore, we modified the input variables to seamlessly integrate with the commercial platform's data structure, specifically injecting the {`subject`}, {`topic_title`}, {`context`}, and {`questions`}. The complete prompt template is presented below.

**System Instruction: Dinucu_Prompt (Static Baseline)**

```
You are tasked with being a teacher and helping a student with a
{subject} problem. You must not reveal the answer to the problem to
```

```
the student at any point in time.
Your task is to guide the student to have a complete understanding of
the problem. Even if the student is already able to solve the problem,
you should help them understand and improve the solution so that they
get as high of a grade as possible. If possible, do not respond with
overly long responses to the student.
Here is the {subject} problem:
### Topic Title: {topic_title}
### Topic Context: {context}
### Question data: {questions}
```

## B.4 Linguistic Style Analysis Prompt

This prompt is utilized by the high-capacity LLM (GPT-5-mini) to extract stylistic features from authentic student dialogue samples ($\mathcal{D}_{chat}$).

**System Instruction: Style Analyst**

```
# ROLE
You are a linguistic analyst specializing in adolescent communication
patterns.
# TASK
Analyze the writing style of these Dutch high school student replies:
[Sampled_Chat_Logs]
# OUTPUT REQUIREMENTS
Describe the tone, grammar proficiency level, typical sentence
structures, and common linguistic errors in one concise paragraph.
This summary will be used to condition a student simulator.
```

## B.5 Student Simulator Persona (Unmotivated Profile)

This section details the behavioral logic for the *Unmotivated* student. Unlike static personas, our simulator employs a dynamic prompt injection mechanism to enforce realistic brevity and off-topic behaviors.

**System Instruction: Unmotivated Student Persona**

```
# ROLE
You are a distracted and unmotivated Dutch high school student learning
{topic_title} in {subject}.
# BEHAVIORAL TRAITS
- Low Patience: Frequently ignore questions or provide random answers.
- Linguistic Realism: Use Dutch-English code-switching and heavy slang.
- Style Summary: {style_summary}
```

### Dynamic Constraint Injection (Latent Thoughts)

To prevent the LLM's natural tendency toward verbosity, the system appends a random "inner monologue" hint to each teacher utterance before it reaches the student simulator.
Examples of injected hints (randomly sampled):
- [Think: you are confused; reply in one short line.]
- [Think: ignore the question and talk about something else.]
- [Think: you give a wrong answer. Reply in one short line.]

# Appendix C

# Score Modeling and Evaluation Rubrics

This appendix provides the complete evaluative framework used by the LLM-based score model. Section C.1 presents the structural prompt template, while Section C.2 details the 14 pedagogical criteria used for binary feature extraction.

## C.1 Pedagogical Evaluator Prompt

**System Instruction: score Evaluator**

```
You are an expert educational conversation evaluator. Using the
guidelines provided in the next section, evaluate the following chat
transcript.
For each criterion, think step by step and make a binary True/False
decision based only on clear evidence in the transcript. If the
criterion cannot be confirmed confidently, mark it as False. Do not
infer user intent or emotion that is not explicitly expressed.
Return ONLY a JSON object with True/False for each criterion:
{
"answer_2_questions": boolean,
... (14 features) ...
}
### Dialogue Trajectory:
[Full Transcript]
```

## C.2 Pedagogical Labeling Guidelines (14 Criteria)

The following rubrics define the classification boundaries for the LLM evaluator. These criteria ensure that the score signal is grounded in objective pedagogical markers.

1. **Answer $\geq$ 2 questions**: The user answers two or more questions posed by the assistant about the main topic. (False if fewer than two or off-topic).

2. **Ask $\geq$ 2 on-topic questions**: The user asks two or more questions (why, how, what, when) connected to the topic. (Exclude generic greetings).
3. **Interact $>$ 3 times**: The user sends four or more substantive messages containing reasoning or topic-related inquiry.
4. **Positive social exchanges**: The user expresses positivity (e.g., "thank you", emojis) at least twice.
5. **Answers mostly correct**: The user's responses align with explanations in most cases with few clear mistakes.
6. **Correct within 2 turns**: The user provides a correct response within two attempts at least 75% of the time when prompted.
7. **Shows understanding**: Relevant responses and follow-up questions demonstrate comprehension without repeated confusion.
8. **Shows curiosity**: Asks at least one question that goes beyond basic requirements (explores "why" or "how").
9. **Justifies mistakes**: After an error, the user either reflects on the reasoning or provides a corrected answer later.
10. **Assistant on topic**: The assistant remains focused on the learning goal throughout the interaction.
11. **Assistant scaffolding**: The tutor offers progressive, multi-turn guidance and adjusts help level when the student struggles. (False if gives full answers immediately).
12. **Assistant diagnoses**: The assistant identifies specific mistakes and provides tailored clarifications.
13. **Assistant balances**: The assistant alternates between explaining and prompting, avoiding a monologue-style delivery.
14. **Assistant adapts**: The tutor changes behavior (e.g., more explanation after mistakes) based on student performance.