

# Enhancing Adaptive Learning Using Reinforcement Learning For Real-Time Code Debugging & Feedback

Shantanu R. Mandawkar<sup>1</sup>, Krishna S. Khaple<sup>2</sup>, Netraja C. Mulay<sup>3</sup>, Prakash Kene<sup>4</sup>

*Department of MCA, P.E.S. Modern College of Engineering, Pune, India*

**Abstract-** This paper proposes a novel adaptive learning framework integrating reinforcement learning (RL) and transformer-based natural language processing (NLP) to deliver real-time, personalized, pedagogically rich feedback. Moving beyond static feedback models, the system uses Q-learning and policy optimization to identify student error patterns and adapt responses dynamically. The NLP module, leveraging spaCy and transformer architectures, generates context-specific responses tailored to learners. A four-week controlled experiment with 200 middle school students demonstrated significant improvements in performance, retention, and engagement. The system's scalability, ethical transparency, and user-centered design offer promising directions for intelligent tutoring systems.

## I. INTRODUCTION

Adaptive learning systems aim to tailor educational experiences based on student behavior, performance, and preferences. However, many systems rely on delayed or binary feedback, limiting opportunities for real-time error correction and conceptual development. This paper introduces a next-generation adaptive learning platform that combines reinforcement learning and NLP to offer timely, personalized feedback.

Using Q-learning, the system adjusts feedback based on student interactions. NLP components built on transformer models generate context-aware explanations, reinforcing student understanding. A study with 200 students over four weeks demonstrated improvements in engagement, accuracy, and time-on-task. This paper presents the system's theoretical framework, implementation details, and implications for intelligent tutoring technologies.

## II. LITERATURE REVIEW

The development of adaptive learning systems has transitioned from foundational behaviorist principles to more sophisticated intelligent tutoring

frameworks. While traditional ITS models offer prescribed feedback, they lack responsiveness and nuance. Bloom and Bruner emphasized the benefits of personalized instruction, laying the groundwork for AI-enhanced adaptive systems.

Recent research explores deep reinforcement learning (DRL) to tailor learning paths. Khosravi et al. (2022) showed how DRL adapts content based on mastery levels. Gupta and Majumdar (2021) demonstrated Q-learning's efficacy in dynamic hinting strategies.

Transformer-based NLP models like BERT and GPT have enhanced machines' ability to generate human-like dialogue. Chandak et al. (2020) used these models to create educational agents capable of interpreting student intent and delivering supportive feedback. UX studies also highlight the importance of adaptive interfaces, with real-time feedback and simplified navigation improving student satisfaction. Despite progress, few systems integrate RL, NLP, and UX into a unified feedback model. This study will lower that gap with a cohesive, real-time feedback system.

## III. PROBLEM STATEMENT

Most adaptive learning platforms offer binary feedback, which lacks actionable guidance. Phrases like "Correct" or "Incorrect" do not inform students why an answer is wrong or how to fix it. Furthermore, most systems fail to leverage reinforcement learning and NLP jointly, leading to delayed, generic feedback.

This research addresses the need for an integrated feedback system that:

- Detects learning errors in real time
- Optimizes responses via RL (reward shaping, policy learning)
- Delivers personalized feedback in natural language using transformer-based NLP.

The objective is to enhance cognitive engagement, learning retention, and educational equity through responsive, context-aware feedback.

#### IV. METHODOLOGY

##### 1) Theoretical and Technical Framework

The system integrates RL, transformer-based NLP, and cognitive theory-aligned feedback delivery:

- **Reinforcement Learning:** The Q-learning algorithm maps student states (e.g., correctness, response time) to optimal feedback actions. Reward shaping aligns with pedagogical goals, while policy optimization adapts strategies over time.
- **Transformer-Based NLP:** Built with spaCy 3.0 and DistilBERT embeddings, the model classifies error types and generates context-aware responses. Templates are dynamically filled using slot-filling and sentiment balancing techniques.
- **Cognitive Theory:** Feedback aligns with Constructivist and Cognitive Load Theory principles—delivered in real-time to support conceptual reconstruction without overwhelming the learner.

##### 2) Data Collection & Preprocessing

Data was collected from 200 middle school students over four weeks in mathematics. Logged metrics included response accuracy, time per question, retry attempts, and topic identifiers.

Preprocessing steps:

- Removed outliers (e.g., >2-minute responses)
- Normalized numerical features
- Encoded categorical variables
- Annotated error patterns via teacher review.

##### 3) Experimental Design

Two groups were compared:

Group	Size	Feedback Type
Control	100	Binary (Correct/Incorrect)
Experimental	100	NLP-based intelligent feedback

Pre/post-tests assessed learning outcomes. System logs captured engagement and retry behavior.

##### 4) Statistical Analysis

- Paired t-tests (pre/post comparisons)
- Independent t-tests (between groups)
- Cohen’s d for effect size (d = 1.6: large effect)
- 95% confidence intervals,
- Visualizations: box plots, radar charts, and score distributions.

#### V. IMPLEMENTATION

##### 1) System Architecture

The platform follows a modular micro service architecture with four key modules:

- **Interaction Logger:** Tracks student actions (attempts, retries, timestamps)
- **RL Engine:** Applies Q-learning to optimize feedback policies
- **NLP Generator:** Uses spaCy + Transformer for natural language output
- **Frontend Interface:** Web-based UI showing feedback via chat bubbles, visuals, and audio.

##### 2) Sample Code Logic

A simplified Python snippet demonstrates the reward-based feedback generation process using spaCy and TensorFlow:

```
python

nlp = spacy.load("en_core_web_trf")
model = tf.keras.Sequential(...)
model.compile(optimizer='adam', loss='mse')
model.fit(X, y)

def generate_feedback(topic, error_count):
    reward = model.predict([[error_count]])[0][0]
    doc = nlp(topic)
    return f"✔ Good job!" if reward > 0.6 else f"✘ Let's revisit {doc.text}."
```

The code predicts reward scores and generates feedback accordingly, simulating dynamic educational guidance.

#### VI. RESULTS AND DISCUSSION

##### 1) Learning Gains

The experimental group improved by 15.2 points (SD = 6.1), versus 5.4 (SD = 5.9) in the control group. A t-test confirmed statistical significance:  $t(198) = 10.44, p < 0.001$ .

- Cohen’s d = 1.6 (large effect)
- 95% CI: [7.85, 11.94]

These gains indicate the RL-NLP system enhances learning more effectively than binary feedback systems.

##### 2) Engagement Metrics

Metric	Experimental	Control
Avg. Time-on-Task	18.4 min	12.3 min

Retry Rate	4.8	2.1
Voluntary Extensions	46%	19%

Higher engagement in the experimental group suggests students were more motivated to reattempt and persist, reinforcing learning behaviors.

### 3) Error Correction Behavior

63% of missed concepts were corrected in the experimental group on follow-up questions, compared to 34% in the control group, implying improved retention and concept transfer.

4) Qualitative Feedback Students described the feedback system as:

- “Like a teacher helping me,”
- “Told me what I did wrong,”
- “Motivated me to try again.”

These responses support the value of empathetic, personalized feedback in reducing anxiety and enhancing resilience.

### 5) Analysis

- RL Optimization: The system's success lies in ability to adjust feedback type (e.g., hints, praise) using Q-values and policy optimization.
- NLP Personalization: Transformer-based NLP generates nuanced, student-specific language that facilitates comprehension and reflection.
- Cognitive Alignment: The approach supports real-time schema construction and avoids overload by adjusting feedback complexity to the learner's proficiency.

## VII. CONCLUSION & FUTURE WORK

This study demonstrates that a reinforcement learning and NLP-based feedback system significantly improves learning outcomes, engagement, and conceptual retention. The integration of real-time, context-sensitive feedback supports deeper learning and higher persistence among students.

Future directions include:

- Scaling the system for mobile/cloud environments
- Expanding to other subjects and age groups
- Addressing NLP bias through broader training data
- Longitudinal studies to assess lasting impact

This integrated model offers a robust foundation for the next generation of intelligent tutoring systems.

## REFERENCES

- [1] Atkinson, R. C.. The role of the computer in teaching reading. Proceedings of the Invitational Conference • • • • • on Testing Problems. <https://file.eric.ed.gov/fulltext/ED080541.pdf>
- [2] Ausubel, D. P. . Theory and problems of child development. Grune & Stratton.. <https://psycnet.apa.org/doiLanding?doi=10.1037/2Fh0082089> Bloom, B. S.. The role of the educational sciences in curriculum development. International Journal of Educational Sciences. <https://scispace.com/pdf/benjamin-s-bloom-s-contributions-to-curriculum-instruction-4ukhiu7irj.pdf>
- [3] Bruner, J. S. . The course of cognitive growth. American Psychologist, 19(1), 1–15. <https://doi.org/10.1037/h0044160>
- [4] Explosion AI. (2021). spaCy: Industrial-strength natural language processing (Version 3.0) [Software]. <https://spacy.io/>
- [5] Bloom, B. S. (1984). The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring. Educational Researcher, 13(6), 4– 16. <https://doi.org/10.3102/0013189X013006004>
- [6] Bruner, J. (1960). The Process of Education. Harvard University Press.
- [7] Sweller, J. (2020). Cognitive Load Theory and Educational Technology. Educational Technology Research and Development, 68, 1-16, <https://eric.ed.gov/?id=EJ1243787>
- [8] Khosravi, H., Kitto, K., & Waters, Z. (2022). Deep Reinforcement Learning for Adaptive Learning Pathways. DOI:10.48550/arXiv.2004.08410
- [9] Arthur C. Graesser, Kurt Vanlehn, Carolyn Penstein Rosé & Pamela W. Jordan. Intelligent Tutoring Systems with Conversational Dialogue DOI:10.1609/aimag.v22i4.1591
- [10] Stefan Popenici, Sharon Kerr. Exploring the impact of artificial intelligence on teaching and learning in higher education. DOI:10.1186/s41039-017-0062-8
- [11] Jelena Ilić, Mirjana Ivanovic, Aleksandra Klasnja Milicevic. The intelligent tutoring impact of systems and artificial intelligence on students' motivation and achievement in STEM education: A systematic review. DOI:10.5937/JESMAC2402005I

- [12] Kay K. Seo and Scott Gibbons. Learning Technologies And User Interaction. <https://api.taylorfrancis.com/content/books/mo no/download?identifierName=doi &identifierValue=10.4324/9781003089704&t ype=googlepdf>