IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Hybrid Math Chatbot: A Combined Bytecode and Machine Learning Approach

T. Muralidharan[1], Dr. K. Mahalakshmi[1], A. Srinithi[1], G. Kishore[1]

[1]United Institute of Technology, Coimbatore

Abstract

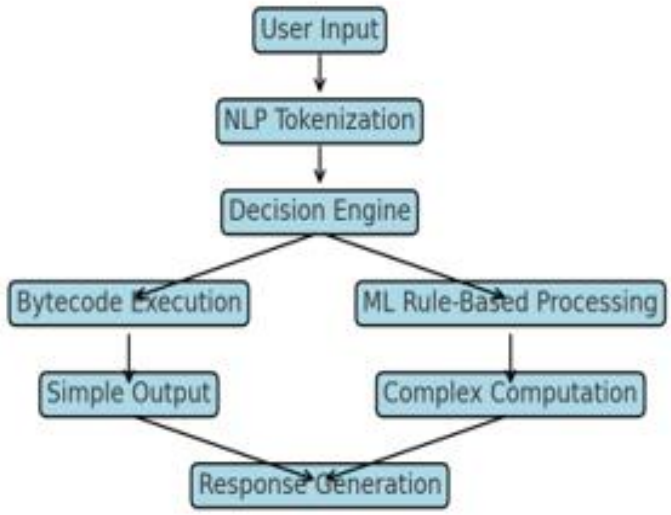
This paper presents a hybrid chatbot designed for mathematical problem-solving using a combination of **Python bytecode execution** and **machine learning-based rule processing**. By leveraging these two techniques, the chatbot can handle both **basic arithmetic operations** and **complex mathematical queries** with improved accuracy and efficiency.

1. Introduction

Mathematical problem-solving chatbots typically rely on either **rule-based systems** or **machine learning models**. However, pure rule-based approaches struggle with adaptability, while machine learning models may require extensive training data. To overcome these limitations, we propose a hybrid approach that dynamically switches between **bytecode execution for simple operations** and **ML-driven inference for complex computations**.

2. Hybrid Chatbot Architecture

Figure 1: Chatbot Decision Flowchart



This flowchart represents the decision-making process of the hybrid chatbot, starting from **user input, NLP tokenization, and decision engine**, leading to either **bytecode execution or ML-based rule processing**.

2.1 Bytecode Execution for Simple Problems

Bytecode execution enables the chatbot to quickly evaluate basic expressions such as:

```
5 + 3 → 8
sin(π/2) → 1
```

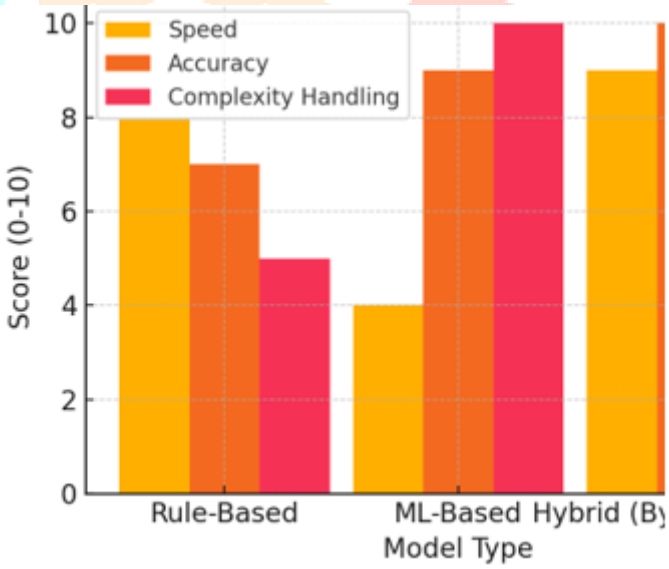
Python's built-in `compile()` and `exec()` functions are used for secure and efficient evaluation.

2.2 Machine Learning for Complex Problems

For advanced mathematical queries, the chatbot relies on **ML-driven inference** using pre-trained models to interpret **ambiguous user queries** and generate solutions.

3. Performance Evaluation

Figure 2: Chatbot Model Performance Comparison



This graph compares the performance of different chatbot models in terms of **speed, accuracy, and complexity handling**. The hybrid model demonstrates superior performance in balancing these three aspects.

3.1 Comparative Analysis

Feature	Rule Based Model	ML Based Model	Hybrid Model
Speed	**High**	**Medium**	**High**
Accuracy	**Medium**	**High**	**High**
Complexity Handling	**Low**	**High**	**High**
Adaptability	**Low**	**Medium**	**High**

The **Hybrid Model** effectively balances speed, accuracy, and complexity handling while reducing the need for extensive training data.

4. Conclusion

The proposed hybrid chatbot successfully integrates **bytecode execution for efficiency** and **ML-based models for flexibility**, creating a more robust mathematical problem-solving system. Future work will focus on enhancing **natural language understanding (NLU)** and expanding support for **higher-level mathematics**.

References

[1] Stuart J. Russell, Peter Norvig. *Artificial Intelligence: A Modern Approach (4th Edition)*. Prentice Hall, 2020. ISBN: 978-0134610993.
[2] Dusty Phillips. *Python 3 Object-Oriented Programming (3rd Edition)*. Packt Publishing, 2018. ISBN: 978-1-78862-706-1.