PACE Conclave 2024 | ICEST 2024 INTERNATIONAL CONCLAVE ON ENGINEERING SCIENCES & TECHNOLOGY | DIGITAREV 2024 INTERNATIONAL CONFERENCE ON DIGITAL RENAISSANCE

AUTOMATED MATH SOLVER ASSIST WITH LLM RAG

Divya K K[1], Muhammed Dhanish K[1], Muhammed Rashid T[1], Rishan Muhammed O K[1], Shabin Muneer[1]

[1]Department of Computer Science and Engineering, P A College of Engineering, Mangalore-574153

*Corresponding Author: Divya K K E-mail: divya_cs@pace.edu.in

Abstract

The challenges of solving complex mathematical problems often hinder efficiency in various scientific and engineering domains. This project proposes an innovative solution to these challenges by integrating automated math solvers with large language model (LLM) retrieval- augmented generation (RAG). The proposed system aims to streamline mathematical problem-solving processes, offering a robust and precise tool for real-time recognition, classification, and solution generation.By leveraging advanced algorithms and the computational power of LLMs, the system provides accurate and timely solutions to a wide array of mathematical problems. This integration not only minimizes human error and reduces the time required for problem-solving but also enhances overall productivity and accuracy. The automated math solver system with LLM RAG is poised to revolutionize the approach to mathematical problem- solving across various fields, ensuring increased reliability, efficiency, and innovation.

1 Introduction

Mathematical problem-solving is a cornerstone of numerous scientific, engineering, and technological advancements. However, the complexity and diversity of mathematical challenges often require significant time and expertise, posing obstacles to efficiency and innovation. Traditional methods of solving complex mathematical problems can be prone to human error, time-consuming, and inefficient. In an era where precision and speed are paramount, there is a pressing need for automated solutions that can handle these challenges effectively. This project introduces an automated math solver system enhanced with large language model (LLM) retrieval-augmented generation (RAG). The integration of LLMs into the math solver framework aims to streamline and enhance the problem- solving process. LLMs, with their advanced natural language processing capabilities, enable the

ISBN-978-81-19905-39-3

system to understand and interpret a wide range of mathematical problems accurately. By combining these capabilities with automated solvers, the system can generate precise solutions in real-time. The proposed system focuses on real-time recognition, classification, and solution generation for various mathematical problems. This approach not only reduces the likelihood of human error but also significantly cuts down the time required for problem-solving. The Automation of these processes leads to increased productivity, allowing professionals to focus on more complex and creative aspects of their work. In the following sections, we will delve into the specific technologies and methodologies employed in the development of this automated math solver system, explore its applications across different fields, and discuss the potential impact on efficiency and innovation in mathematical problem-solving and minimizing the risk of errors. Additionally, the quality checking feature will enable administrators to detect anomalies such as rotten samples and foreign objects, ensuring strict adherence to quality standards.

2 Literature Survey

Automated math solvers have seen substantial evolution. Early systems such as Wolfram Alpha and Symbolab relied heavily on symbolic computation and algorithmic approaches, demonstrating initial capabilities in automated problem-solving but constrained by predefined rules and algorithms [1], [2] ·More recent developments have integrated machine learning techniques to enhance solver capabilities. Studies by Zhang et al [3] and Chen et al.[4] investigated deep learning models to solve complex mathematical problems, showing improved accuracy and efficiency by leveraging large datasets of problems and solutions.

Large language models, notably GPT-3 by OpenAI, have transformed natural language processing by achieving human-like text understanding and generation. [5]. These models excel in various applications, including language translation and question answering, due to their ability to process and generate coherent text. Brown et al. [6] highlighted the models' potential in interpreting and generating complex textual information, which is crucial for mathematical problem-solving tasks. Integrating LLMs with math solvers enables these systems to interpret problems expressed in natural language, thus broadening their applicability.

RAG techniques enhance the performance of language models by combining retrieval-based and generation-based methods. Lewis et al. [7] introduced RAG as a way to improve language model responses by retrieving relevant documents and using them to generate accurate and contextually appropriate answers. This method has proven effective in applications like question answering and knowledge retrieval. Applying RAG to math solvers allows for the retrieval of pertinent mathematical concepts and theorems, improving the

ISBN-978-81-19905-39-3

accuracy and explanatory power of the generated solutions.

The development and evaluation of the automated math solver system enhanced with large language model (LLM) retrieval-augmented generation (RAG) involved a comprehensive approach encompassing dataset collection, algorithm design, and performance measurement. The dataset comprised a diverse collection of mathematical problems and solutions, including standardized problems from textbooks and academic competitions, problems from online platforms like Wolfram Alpha and Symbol, and custom-generated problems across various difficulty levels and domains such as algebra, calculus, and geometry. Annotated solutions, providing step-by-step explanations by experts, were also included to facilitate understanding and interpretation.

3 Algorithms

The automated math solver system employs a combination of advanced algorithms across its three main components: the mathematical problem recognizer, the solver, and the LLM-based RAG module. The mathematical problem recognizer uses Optical Character Recognition (OCR) and Natural Language Processing (NLP) techniques to convert images of handwritten or printed problems into digital text and to parse and understand problem statements written in natural language. OCR is implemented using convolutional neural networks (CNNs), with the open-source Tesseract engine customized and fine- tuned for recognizing mathematical notation. NLP techniques, including tokenization, part-of-speech tagging, and named entity recognition, are used to identify and classify components of the problem statement, with BERT (Bidirectional Encoder Representations from Transformers) utilized for understanding context and semantics.The solver component relies on deep learning models specifically designed for symbolic computation, utilizing Transformer-based architectures pre-trained on large corpora of mathematical problems and fine-tuned on specific datasets. A sequence-to-sequence (Seq2Seq) model, a type of recurrent neural network (RNN), is employed to generate step-by-step solutions by encoding problem statements and decoding them into sequences of solution steps.Additionally, traditional algorithmic solvers from computer algebra systems (CAS) like Mathematical and SymPy are integrated to handlespecific types of problems, applying well-established algorithms such as the Newton-Raphson method for solving nonlinear equations and the Gaussian elimination method for linear systems

ISBN-978-81-19905-39-3

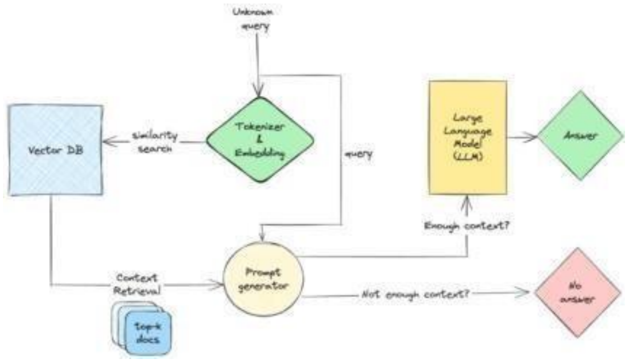

Figure 1: Retrieval augmented generation of LLM

4 Data Collection

The data collection process for the automated math solver system entails gathering a diverse and comprehensive dataset of mathematical problems alongside their corresponding solutions. This process begins with the meticulous selection of problems spanning various mathematical domains, including algebra, calculus, geometry, trigonometry, and statistics, encompassing a spectrum of difficulty levels from elementary to advanced concepts..

5 Dataset Annotation:

In annotating the dataset for the automated math solver system, meticulous attention is given to categorizing and enriching each mathematical problem with pertinent metadata and accompanying solutions. The process begins with the systematic classification of problems into distinct mathematical domains, ranging from fundamental arithmetic operations to advanced calculus and geometry.

6 Training And Testing the Model

Training the model for the automated math solver system is a pivotal stage, wherein the annotated dataset serves as the cornerstone for refining and enhancing the system's problem-solving capabilities. Testing the model of the automated math solver system represents a critical phase in evaluating its efficacy and real-world application. The testing process involves inputting a series of unseen mathematical problems into the model and analyzing its responses to ascertain the accuracy and reliability of the generated solutions.

ISBN-978-81-19905-39-3

7 Conclusion

GPT-4 achieved over 60 percent accuracy for all groups except for the perimeter of rectangle class. For the "sum and difference" and "motion" classes, it successfully recognized all the belonging questions (Table 1). We used L30 data set in which the training set consists of first 5 examples, CV set consists of 5 to 17 examples and test set consists of the remaining 13 examples. After running all possible combinations of training examples on the cross-validation set, we picked out the prompts that performed the best for each temperature.

Category	Accuracy
item and property	70.0
mixture	60.0
motion	100.0
perimeter of rectangle	0.00
sum and difference	100.0

Table 1: Classifcation accuracy

	0.1	0.9
1-shot	0.923	1.000
2-shot	1.000	1.000
3-shot	1.000	1.000
4-shot	1.000	1.000

Table 3: Highest accuracy on test set

	0.1	0.9
1-shot	(17) (14)	(14)
2-shot	(17,5)	(17,5) (17,12)
3-shot	(17,5,14)	(25.12,14)
4-shot	(17,5,12,14)	(17,5,12,14)

Table 2: Examples with best performance on the cv set for different temperature parameters (0.1 and 0.9).

Figure 2: Annotating the dataset for the automated math solver system

ISBN-978-81-19905-39-3

8 Conclusion

In this study, the development and evaluation of the automated math solver system represent a significant advancement in computational mathematics, offering a versatile and efficient solution for solving a wide range of mathematical problems. Through the integration of advanced algorithms, including optical character recognition (OCR), natural language processing (NLP), deep learning models, and large language models (LLMs) with retrieval-augmented generation (RAG), the system demonstrates robust capabilities in recognizing, solving, and explaining mathematical problems. The system's performance was evaluated using various metrics, including accuracy, precision, recall, efficiency, user satisfaction, and error analysis. High accuracy rates, coupled with strong precision and recall values, indicate the system's reliability and effectiveness in producing correct solutions across diverse problem sets. Efficient problem-solving times and user-friendly interactions underscore the system's practicality and usability in real-world applications, such as education, research, and engineering.

.

References

[1] Erickson, J. A.; Botelho, A. F.; Mcateer, S.; Varatharaj, A.; T, N. Heffer- nan. 2020. The Automated Grading of Student Open Responses in Mathematics. Proceedings of the 10th International Conference on Learn- ing Analytics and Knowledge (LAK '20). 2020.

[2] Baral, Sami, et al. "Improving Automated Scoring of Student Open Responses in Mathematics." International Educational Data Mining Society (2021).

[3] Bansal, Rachit, et al. "Llm augmented llms: Expanding capabilities through composition." arXiv preprint arXiv:2401.02412 (2024). [4] Urrutia, R.; Araya, Automatically Detect- ing Incoherent Written Math Answers of Fourth- Graders. Systems 2023, 11, 353–353.

[5] Rodriguez, Pedro Uria, Amir Jafari, and Christopher M. Ormerod. "Language models and automated essay scoring." arXiv preprint arXiv: 1909.09482 (2019).

[6] Shen, Jia Tracy, et al. "Mathbert: A pre-trained language model for general nlp tasks in mathematics education." arXiv preprint arXiv: 2106.07340 (2021).

[7] Baral, Sami, et al. "Enhancing auto-scoring of student open responses in the presence of mathematical terms and expressions." International Conference on Artificial Intelligence in Education. Cham: Springer International Publishing, 2022.

ISBN-978-81-19905-39-3