AI-Powered Mathematical Olympiad Problem Solver

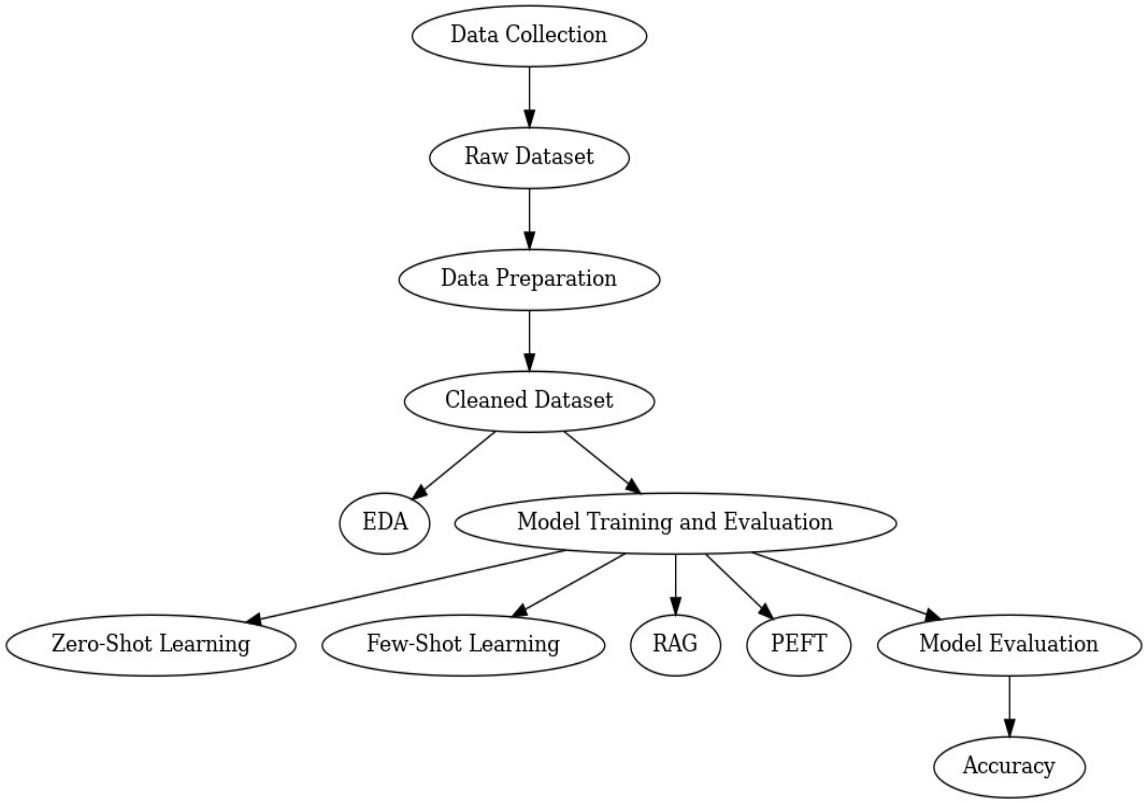
Progress, Methods, and Future Directions

Introduction

- **Objective**:
  - **Goal**: Develop a model that accurately solves mathematical problems from the AI Mathematical Olympiad competition.
  - **Benchmark**: Exceed the performance of the current Gemma 7B benchmark of 3/50 and achieve at least 20% accuracy.
- **Key Components**:
  - **Problem Interpretation** : Develop a system that accurately interprets and understands natural language descriptions of Math Olympiad problems interpretation process.
  - **Mathematical Reasoning** : Implement algorithms to process and reason through mathematical concepts, theorems, and logic.
  - **Evaluation**: Establish a robust evaluation mechanism to assess the accuracy and reliability of the solutions provided by the model.
- **Expected Outcome**:By the end of this project, we anticipate having a robust model capable of solving a wide range of Math Olympiad problems with high accuracy. This model will not only serve as a powerful tool for students and educators but also pave the way for further advancements in AI-driven mathematical problem-solving.

Methodology

- **Data Collection:**
  - Gather: Comprehensive dataset of Math Olympiad problems and solutions.
  - Sources: Web scraping from AIME and Kaggle.
- **Data Preparation:**
  - Clean: Handle missing values, remove duplicates, and standardize formats.
  - Preprocess: Extract numerical answers and ensure data consistency.
- **Exploratory Data Analysis (EDA):**
  - Analyze: Perform summary statistics, distribution plots, and word clouds.
  - Visualize: Use histograms, word clouds, and n-gram frequencies.
- **Model Training and Evaluation:**
  - Techniques: Zero-Shot, Few-Shot, RAG, and PEFT.
  - Evaluate: Use accuracy



Dataset Collection

- **Sources**
  - The dataset has been compiled from reputable sources to ensure a diverse and comprehensive collection of Math Olympiad problems:
  - **Kaggle**: The dataset includes problems and solutions contributed by the Kaggle community, ensuring a wide range of difficulty levels and problem types.
  - **American Invitational Mathematics Examination (AIME):** Problems from AIME provide a high standard of difficulty and are essential for training the model to handle advanced mathematical concepts and techniques.
- **Focus**:
  - Problems and solutions formatted in LaTeX.

Data Preprocessing

- **Data merge**: Combined datasets into a single comprehensive dataset.
- **Normalization**: Standardized formatting and notation across all records to prevent inconsistencies that might confuse the model.
- **Question Filtering:** Selected questions with integer answers, discarding those without.
- **Answer Extraction:** Retrieved integer answers from the solutions for questions where answers were missing or not provided.
- **Cleaning:** Removed extraneous information, such as typographical errors and irrelevant content, to maintain the data's quality and relevance.

Dataset Structure

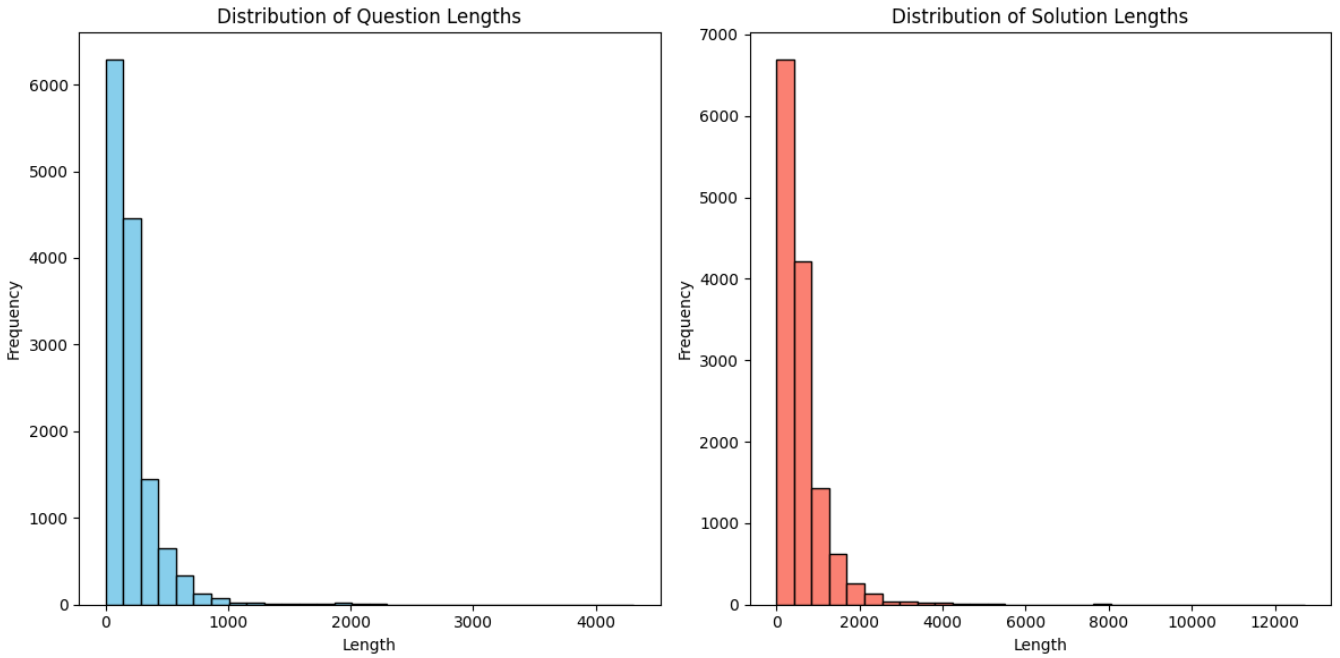
- The training dataset consists of approximately 9000 records, each containing three main components:
- **Questions:** These are the Math Olympiad problems presented in natural language. The questions cover various mathematical topics, including algebra, geometry, number theory, and combinatorics. Each question is designed to challenge the problem-solving abilities and conceptual understanding of the solver.
- **Solutions:** This column contains detailed solutions to the corresponding questions. The solutions are written in a step-by-step manner, illustrating the logical and mathematical reasoning required to arrive at the correct answer. These solutions are critical for training the LLM to understand the process of solving complex mathematical problems.
- **Answer**: The final integer answer

Exploratory Data Analysis (EDA)

- **Basic Information**: Total entries, unique questions, solutions, and answers.
- **Statistics**: Mean, standard deviation, minimum, and maximum lengths of questions and solutions.
- **Visualizations**: Histograms and word clouds.

EDA: Contd.

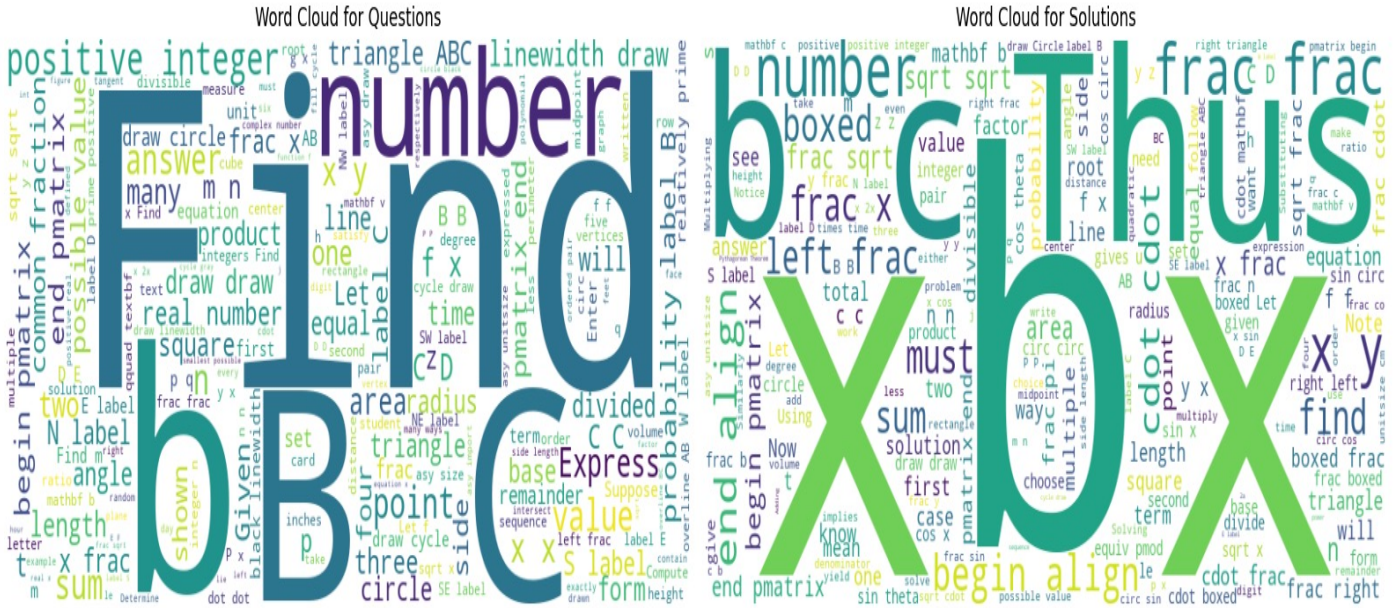
Distribution of question lengths and solution lengths.



- **Question Length Distribution:**
  - Right-Skewed: Most questions are short, with a few very long ones.
  - High Frequency: Majority of questions are under 1000 characters.
- **Solution Length Distribution:**
  - Right-Skewed: Most solutions are concise, with some very long ones.
  - High Frequency: Majority of solutions are under 2000 characters.

EDA: Contd.

Common terms in the questions and Common terms in the solutions.



- **Questions:**
  - Common Terms: “Find,” “number,” “integer,” “positive,” and “real” are the most frequent words.
  - Focus: Emphasis on finding numbers, integers, and specific values.
- **Solutions**:
  - Common Terms: “Thus,” “find,” “number,” “frac,” and “cdot” are the most frequent words.
  - Focus: Emphasis on explaining solutions with mathematical terms and operations.

Methods and Accuracy on 10 Complex Olympiad Problems

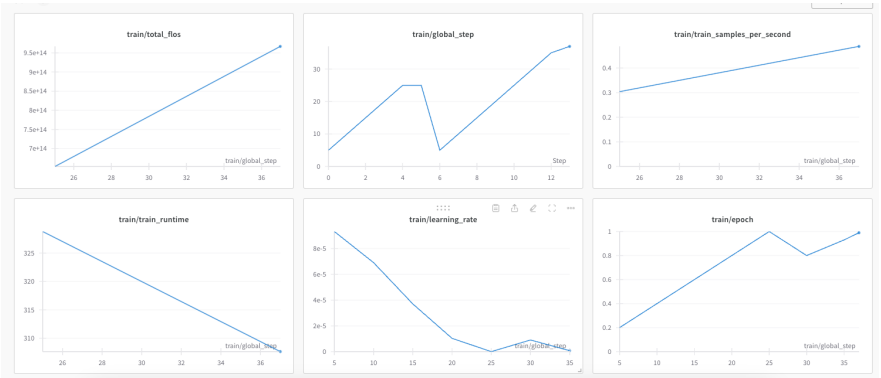
- Z**ero-Shot Learning:**
  - **Description**: Uses a pre-trained model without additional training on specific examples.
  - **Accuracy**: 0%
  - **Note**: The steps were correct in most, but the execution were not leading to incorrect result.
- **Few-Shot Learning:**
  - **Description**: Uses a few examples from AIME to guide predictions.
  - **Accuracy**: 0%
  - No improvement, indicating the need for more relevant examples or techniques.
- **Retrieval-Augmented Generation (RAG):**
  - **Description**: Uses BERT and FAISS to retrieve relevant examples and augment input.
  - **Accuracy**: 10%
  - Showed improvement by leveraging retrieved information, resulting in one correct answer and a better reasoning.
- **Reason for Using 10 Data Points:** Using 10 data points allows for quicker iterations (~1.5min/problem) and faster identification of issues due to time constraints. It is practical to start with a manageable number of complex problems, helping to set a baseline before scaling to larger datasets.

Finetuning Deepseekmath - Training

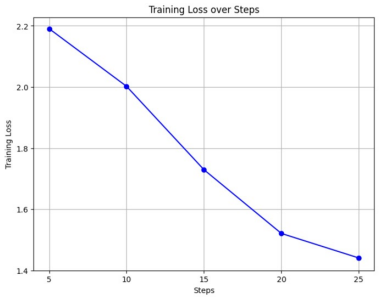
- Finetuned the Deekseek Math model using custom dataset.
- **Dataset**: 100 random samples taken from the Combined datasets from Kaggle and AIME
- **Model Configuration:**
  - Model: Deekseek Math Model (Causal LM with 4-bit quantization).
  - Configurations:
    - BitsAndBytesConfig for 4-bit quantization.
    - LoraConfig: r=20, lora_alpha=40, lora_dropout=0.05.

Finetuning Deepseekmath - Training

- Training Setup:
  - Data Preprocessing: Loaded and cleaned data using Pandas.
  - Training Parameters: Batch size: 1, Gradient accumulation steps: 4, Epochs: 1 Learning rate: 1e-4, Optimizer: Paged AdamW 8-bit, Scheduler: Cosine, Warmup ratio: 0.01
- Training:
  - Used Hugging Face transformers with CUDA.
  - Conducted training using transformers Trainer.
  - W&B is used to track the checkpoints.
- Loss Computation: Training loss computed using Cross-Entropy Loss.



Step	Training Loss
5	2.190600
10	2.002100
15	1.729400
20	1.520900
25	1.440700



Finetuning Deepseekmath - Evaluation

- Evaluation Metric:
  - Accuracy on 10 unseen problems
- Results:
  - Test Accuracy: 10%
- Conclusion: No further improvement in the test accuracy.
- Future improvements:
  - CoT implementation

Self-Consistency Chain Of Thoughts (SC-CoT)

- The reasoning capability of the fine-tuned model has been enhanced in the final step through the application of Self-Consistency in Chain-of-Thought (SC-CoT).
- Instead of selecting the most likely next word or phrase (greedy decoding), self-consistency involves sampling a set of potential reasoning paths from the model's decoder.
- This sampling is crucial as it generates diverse ways the model can think about and solve the given problem.
- The diversity in the sampled paths is typically achieved using different decoding strategies that allow for randomness and exploration of less likely options. We've included the following strategies:
    - Randomly choosing between direct and SymPy based computation:
    - Temperature Sampling: Adjusting the temperature parameter.
    - Top-k Sampling: This method involves sampling from the top k most likely next words or tokens.
- After sampling multiple reasoning paths, these paths are evaluated to determine which conclusions are reached most frequently. The idea is that the correct answer is likely the one that multiple independent reasoning paths agree on.
- SymPy is integrated for calculations and generating reasoning paths for better comprehension and accuracy.

SC-CoT Implementation

- Configuration Settings:
  - n_repetitions: 15 - Number of times the model attempts different reasoning paths for each problem.
  - TOTAL_TOKENS: 2048 - Maximum number of tokens generated for each prompt.
  - TIME_LIMIT: 31,500 - Time limit for processing each problem.
  - MODEL: The fine-tuned DeepSeek math model with memory optimization
- Gradient checkpointing is used manage memory efficiently.
- Quantization is used for optimized model performance using 4-bit precision.
- SymPy code is derived from the model's output, executed locally, and its results are subsequently reintegrated into the prompt for subsequent steps.
- Model output are processed to handle errors and perform retries as necessary.
- Final answers are evaluated through SymPy code execution.

Note: Due to its size exceeding 10GB, this model is not included with the submitted project materials.

Results

- The model is used to solve 10 unseen Olympiad problems
- Accuracy: 10%
- There has been a slight improvement in the model accuracy
- Further analysis of the model's reasoning is required to determine whether improvements are needed in its logical development capabilities or in its execution of those logics.

Conclusion

- **Development Achievements**: The project has successfully engineered an AI model that closely approaches its initial accuracy goals.
- **Integration of Reasoning Techniques**: Beyond the best performing Self-Consistency in Chain-of-Thought (SC-CoT), the project has leveraged a variety of advanced reasoning methods. These include Zero-Shot and Few-Shot learning, Retrieval-Augmented Generation (RAG), as well as Fine tuning each contributing to the model's nuanced understanding and handling of complex mathematical challenges.
- **Reflective Outlook**: While the current outcomes may fall short of high expectations, they underscore the necessity for ongoing enhancements. Future efforts will be directed towards refining the model's precision and broadening its application spectrum to encompass a wider array of problem types and educational settings, thereby enriching its educational impact.

Reference

- AIMO challenge: https://www.kaggle.com/competitions/ai-mathematical-olympiad-prize
- AIME: https://artofproblemsolving.com/wiki/index.php/2018_AIME_I_Problems/Problem_3
- https://dassum.medium.com/fine-tune-large-language-model-llm-on-a-custom-dataset-with-qlora-fb60abdeba07
- https://github.com/ZeusSama0001/RAG-chatbot/blob/main/model.py
- https://www.kaggle.com/code/awsaf49/aimo-kerasnlp-starter

Reference

- https://github.com/asokraju/LangChainDatasetForge/blob/main/Finetuning_Falcon_7b.ipynb
- https://huggingface.co/deepseek-ai/deepseek-math-7b-base
- https://arxiv.org/pdf/2203.11171
- ChatGPT