Overview for

Eedi - Mining Misconceptions in Mathematics

**Competition Overview:**

This competition aims to develop an NLP model that predicts the affinity between distractors in multiple-choice mathematics questions and the underlying misconceptions. Traditionally, each distractor must be carefully matched with a corresponding misconception—an approach that is both time-consuming and prone to inconsistencies. By automating or semi-automating this matching process, the competition seeks solutions that cover known misconceptions while also generalizing to newly emerging ones, thereby reducing manual labeling effort and improving overall educational quality.

The evaluation metric used in this competition is Mean Average Precision @ 25 (MAP@25). Participants can submit up to 25 predicted misconceptions per test instance, and the model is assessed on how accurately and how early (in terms of ranking) it predicts the correct misconception. This metric encourages models to place the correct match as high as possible on the list, reflecting both practicality and effectiveness.

**Algorithm Descriptions:**

This project uses fine-tuned Large Language Models (LLMs) for retrieval and reranking to identify erroneous concepts in mathematics problems. The process is as follows:

1. Data Generation and Model Training
   1. First, gpt-4o is used to generate missing erroneous concepts in the official dataset.
   2. The basic information of each sample is then constructed into training data.
   3. Fine-tuning is performed on Qwen-14b-insrtuct and Qwen-32b-insrtuct models using contrastive learning. During this stage, 4-bit quantized versions of these models are employed, and LoRA is applied to multiple linear modules for training.
2. Retrieval and Model Ensemble
   1. In the retrieval phase, embeddings from three differently trained LLMs (or those with different parameter settings) are concatenated.
   2. Candidate items are then selected through similarity-based retrieval on these combined embeddings.
3. Reranking
   1. Qwen-32b-insrtuct-AWQ is used to rerank the top 25 predictions from the retrieval stage.
   2. Reranking prompts are input into Qwen-32b-insrtuct-AWQ, leveraging its zero-shot capabilities to produce a new ranking. The top 25 results are processed in multiple batches.
   3. Meanwhile, bge-large-en-v1.5 is used to search a specified retrieval library via similarity measures to find the most suitable final results.