Prepared by **Annanahmed Shaikh, Rohan Pratap Reddy Ravula, GROUP 5**

A Novel Fine-Tuning Approach Using Backward Attention Mechanism

Enhancing Math Reasoning in GPT-2 with a Novel Final Attention Head: A Fine-Tuning Approach on OpenMathInstruct-2

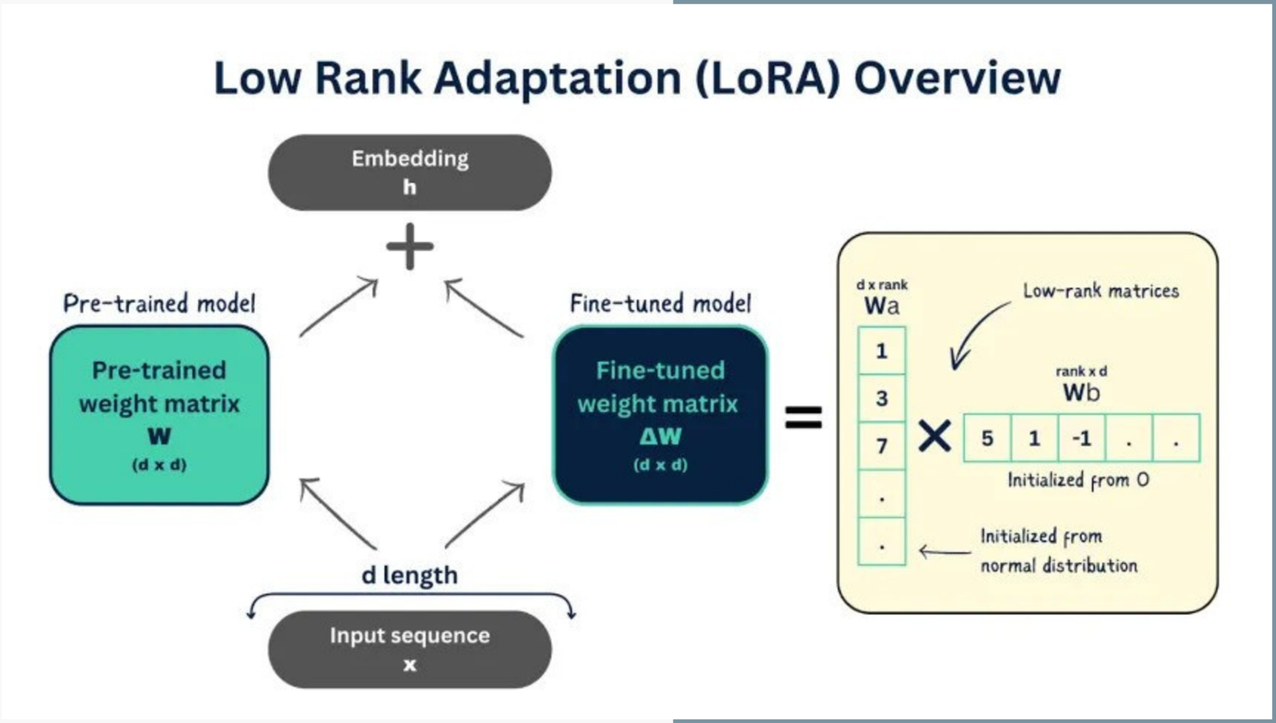
02 April, 2025

Course Name: **Adv Topics in Large Lang Model**
Subject Code: **DATA 6300**

WENTWORTH INSTITUTE OF TECHNOLOGY
Professor Salem Othman

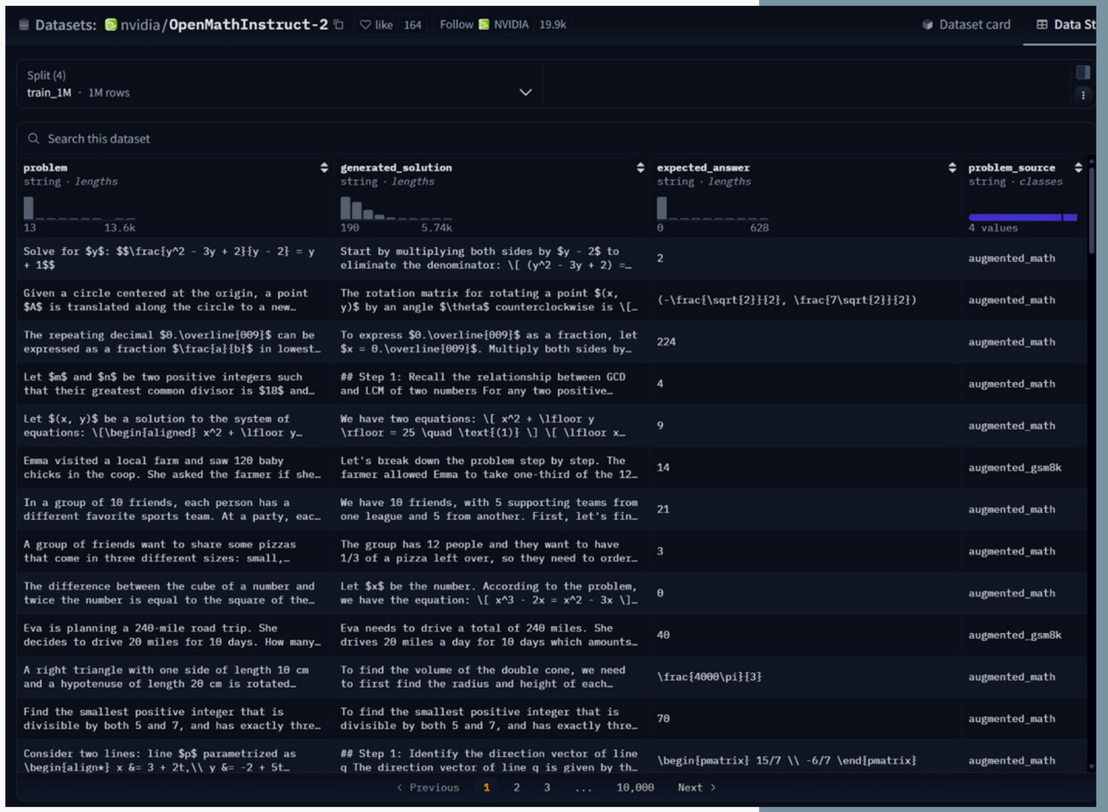
Motivation & Problem Statement

- Traditional token selection methods (greedy, top-k) lack robustness in math-based reasoning.
- Manual tuning (temperature, top-k) required for quality generation.
- Full model fine-tuning is computationally expensive.
- Our Goal: Enhance GPT-2's reasoning capability by replacing the final linear head with a Backward Attention module.
- Inspired by LoRA and QLoRA to maintain efficiency.



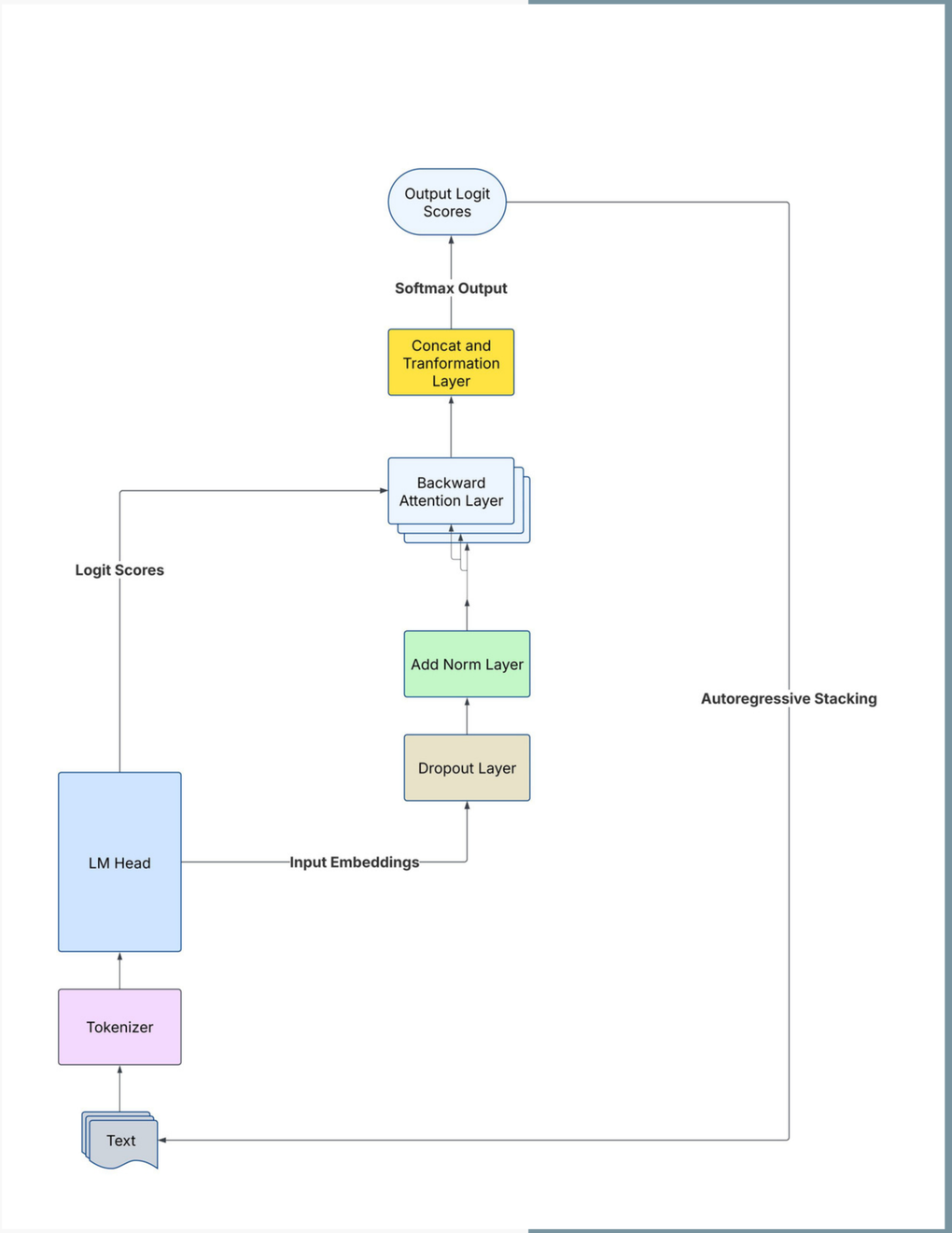
Dataset Overview

- Dataset Used: OpenMathInstruct-2 (by NVIDIA)
- Type: Math reasoning dataset
- Size: 12.6 GB (generated using Llama3.1-405B-Instruct)
- Input: "problem"
- Output: "generated solution" + "expected answer" (merged)
- Format: Supervised Q&A style fine-tuning



Model Architecture

- GPT-2 (Frozen)
- Additional Backward Attention Layer
- Flow: Text Tokenizer Embeddings K, V, Logits Weighted V Query Dot(K,Q) Softmax Output
- Output: Refined probabilities for next-token prediction



Backward Attention Explained

- Step 1: Normalize input embeddings (RMSNorm)
- Step 2: Project to latent space generate Key (K) and Value (V)
- Step 3: Weight V using LM logits Sum Normalize Query (Q)
- Step 4: Dot(Q, K) Softmax Output Probabilities
- Removes need for temperature/top-k tuning
- Adds randomness for robustness in token selection

Pseudo Code

- **Input:**
- IE Input Embeddings (Vocab Size × Embedding Dimension)
- Logits Output logits from frozen GPT-2
- Config Model configuration (dims, flags, device)
- **Procedure:**
- 1. Normalize IE using RMS normalization.
- K IE × WK
- V IE × WV

Compute weighted vector:

- Weighted-V Logits × V // Scale value vectors
- Dot Sum (Weighted-V) / Vocab Size

Project dot product:

- Q Dot × WQ // Query vector

Attention score:

- Attention Q × Transpose(K)

Output:

- Output SoftMax (Attention)
- Output token probabilities

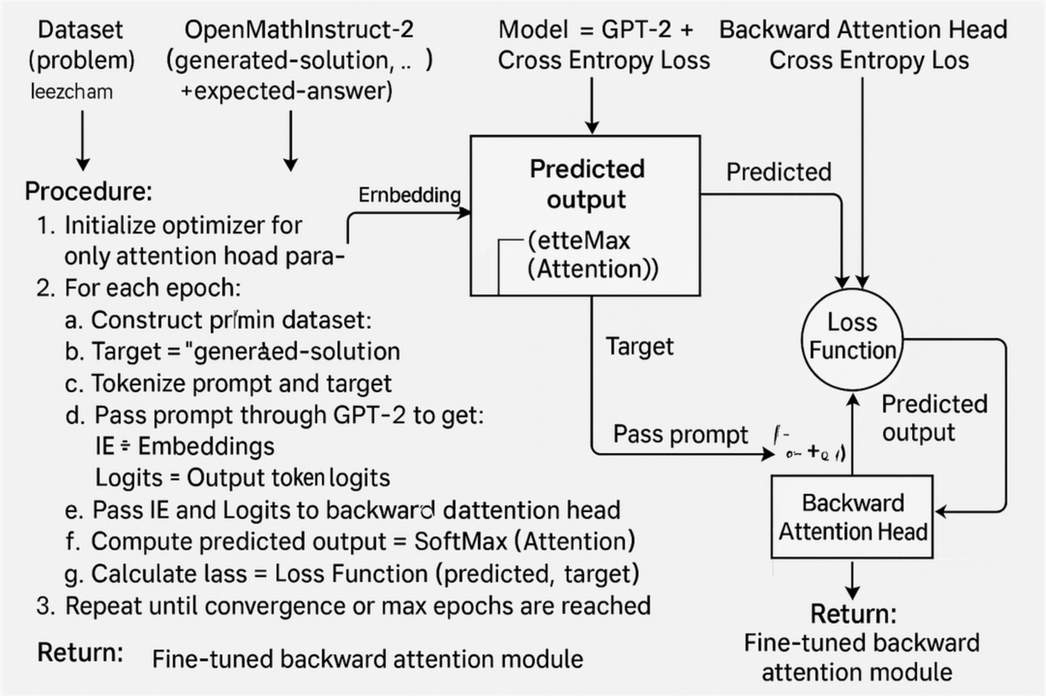
Training Strategy

- Approach: Supervised Fine-Tuning with Teacher Forcing
- Loss Function: Cross-Entropy
- Metrics:
  - Perplexity
  - MSE, RMSE (for numeric answers)
  - Accuracy, Precision, Recall, F1
- Limitation: Training not completed due to hardware constraints
- But full training pipeline + modular architecture is implemented

Due to lack of system hardware, we couldn’t train our model on time.

Evaluation Strategy

- Baselines:
  - GPT-2 fine-tuned on academic topics
  - Llama3.1-405B-Instruct (original dataset generator)
- Expected Gains:
  - Improved token precision in math contexts
  - Better generalization through probabilistic attention
- Metrics Focus: Quantitative (accuracy), Qualitative (reasoning clarity)



Future Work

- Multi-Head/Layered Backward Attention: Better contextual depth
- Dynamic Chunking: Optimize memory + performance
- Non-linear transformations: Add complexity to token interactions
- Transfer Learning: Adapt to small domain-specific datasets
- Real-world deployment: Modular LLM assistants, math tutoring bots
- Error analysis & new metrics: Refine contextual token evaluation
- **Modularity:** This model can be called multiple times for differrent task without disturbing the GPT Model, Hence They can act as the assistant doing different tasks.

Thank you