

---

# **L-Diff: Fast and Efficient Pretraining–Finetuning Framework for Differential Transformer-based Language Models**

---

**AI502 Principles of Deep Learning Final Project**  
**15 Dec, 2025, Team ICCL**

**Yujeong Son**

Department of Electrical Engineering,  
Ulsan National Institute of Science and Technology (UNIST)  
syj4739@unist.ac.kr

**Eunseok Cho**

Artificial Intelligence Graduate School,  
Ulsan National Institute of Science and Technology (UNIST)  
eric754@unist.ac.kr

# Table of Contents

- I. Introduction
- II. Backgrounds
- III. Proposed Method
  - a. Architecture
  - b. Pretrain Method
  - c. Fine-tuning
- IV. Experiments
- V. Conclusion

# Introduction

- Large Language Models (LLMs) keep scaling in parameter count and context length, but efficient and stable training under limited compute remains difficult.
- Differential Transformer (DT) improves attention by suppressing redundant activations and filtering attention noise, enabling more precise context modeling.
- However, DT is harder to train efficiently because it involves dual attention computations that increase memory and compute overhead.

**Goal:** Build a practical framework to make DT **fast and efficient for pretraining** under a **single-GPU** setting.

# Backgrounds

## Standard Attention vs Differential Attention

- Standard scaled dot-product attention computes:

$$\text{Attention}(Q, K, V) = \text{softmax} \left( \frac{QK^T}{\sqrt{d_k}} \right) V$$

- In practice, standard attention can attend to irrelevant context, injecting noise into context aggregation.
- Differential Transformer splits Q and K into two groups and computes two attention maps, then uses their difference to reduce noise.
- This “difference-based” attention is effective, but it can introduce higher memory overhead and training complexity.

## FlashAttention-2

- A major bottleneck in attention is the quadratic memory cost of materializing the  $N \times N$  attention matrix.
- FlashAttention-2 avoids attention matrix materialization by tiling Q/K/V into SRAM-sized blocks and computing attention outputs in a fused manner.
- This reduces memory I/O complexity and enables faster attention computation, especially for long sequences.

# Architecture

## L-Diff

- We propose L-Diff, a fast and efficient pretraining–finetuning framework for Differential Transformer (DT)-based language models.
- L-Diff combines:
- FlashAttention-2–enhanced DT architecture for memory-efficient long-context training, and
- a structured learning pipeline consisting of pretraining followed by two-stage alignment.

## FlashAttention-2 Integrated Differential Transformer

- DT eliminates attention noise by subtracting two attention maps, but this typically increases memory overhead due to dual-head computations.
- We integrate FlashAttention kernels into DT to maximize throughput and memory efficiency.
- Using FlashAttention’s IO-aware algorithm, we reduce frequent HBM↔SRAM read/write operations.
- The differential subtraction logic is computed inside the fused kernel, reducing the need to materialize large  $N \times N$  attention matrices.
- Result: higher training speed and better handling of longer sequences than standard DT implementations.

# Pretrain Method

## Pretraining Setup

- Dataset: WikiText-2
- Objective: Causal Language Modeling (CLM) — predict the next token given the preceding context.
- Target: robust convergence and computational efficiency in a single-GPU environment.

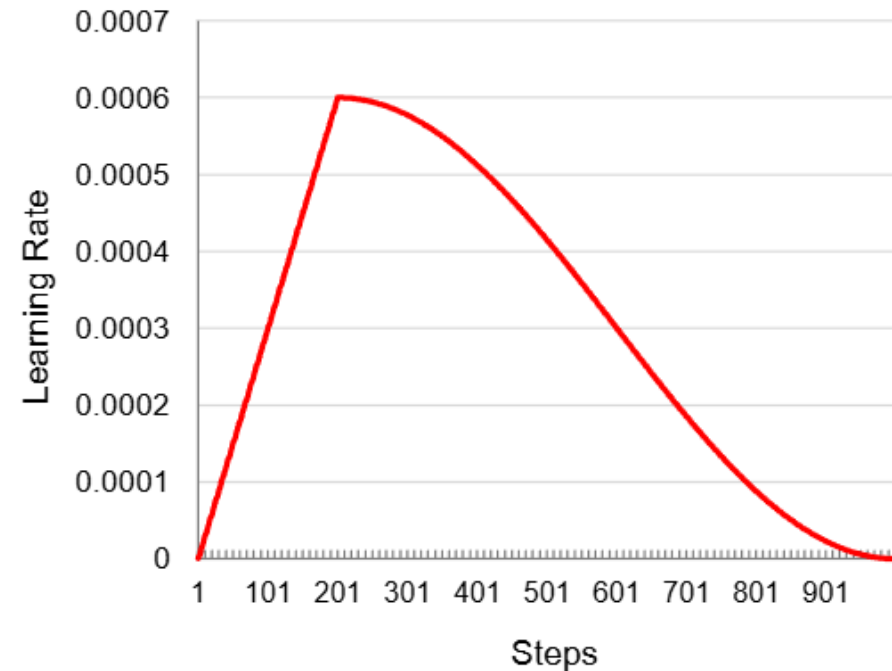
## Pretrain Method 1: Tensor Alignment

- We optimize training throughput through tokenizer–vocabulary alignment and hardware-friendly tensor shapes.
- The tokenizer vocabulary is padded to the nearest multiple of 64, enabling tensor-core-friendly matrix dimensions.
- This reduces overhead in the embedding and output projection layers and improves GPU efficiency.

# Pretrain Method

## Pretrain Method 2: Optimization and Scheduling

- Optimizer: AdamW with weight decay = 0.1 to mitigate overfitting on the relatively small WikiText-2 corpus.
- Scheduler: Cosine Annealing learning-rate schedule with a 200-step linear warmup.
- We use an aggressive peak learning rate ( $6 \times 10^{-4}$ ) to quickly capture dominant linguistic patterns and escape local minima early, then decay for stable convergence.



## 2-Phase Fine Tuning

- **Phase 1: Supervised Fine Tuning (SFT)**

- Dataset: SlimOrca
- LoRA Hyperparameter: **64** rank and alpha to **128**  
This provide a higher capacity for the model to capture and represent intricate task-specific patterns
- Learning Rate: **2e-4**

- **Phase 2: Direct Preference Optimization (DPO)**

- DPO fine-tunes a LLM directly on human preference data (pairs of preferred and rejected responses) without needing a separate reward model or complex reinforcement learning.
- Dataset: Intel/orca\_dpo\_pairs
- LoRA Hyperparameter: maintained to **64** rank and alpha to **128**
- Learning Rate: **5e-6**

# Experimental Results: Pretraining

Category	Specification
GPU	NVIDIA RTX A6000
Parameter Size	159M
Pretraining Dataset	WikiText-2
Training Precision	BFloat16
Max Sequence Length	2048

Table 1: Experimental Environments

Configuration	Training time (m)
Vanilla Transformer	~ 180
DT+nopretrain	~ 60
L-Diff	~ 50

Table 2: Training Time Comparison

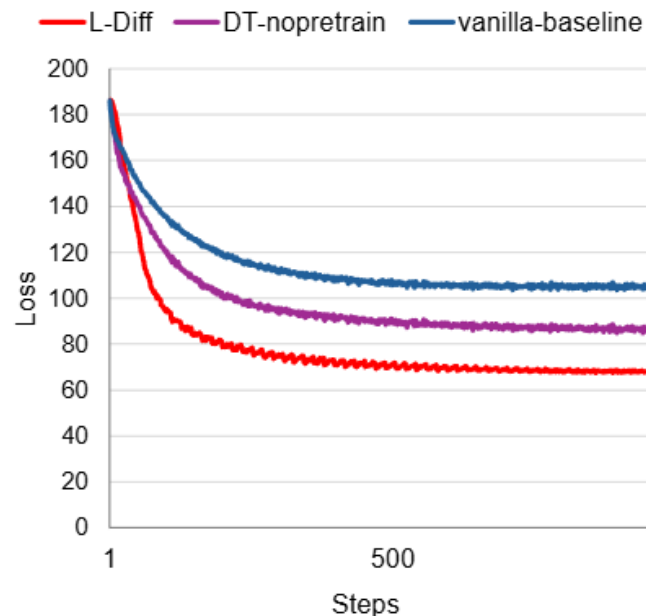


Figure 2: Loss of the first 1000 steps during the pretraining.

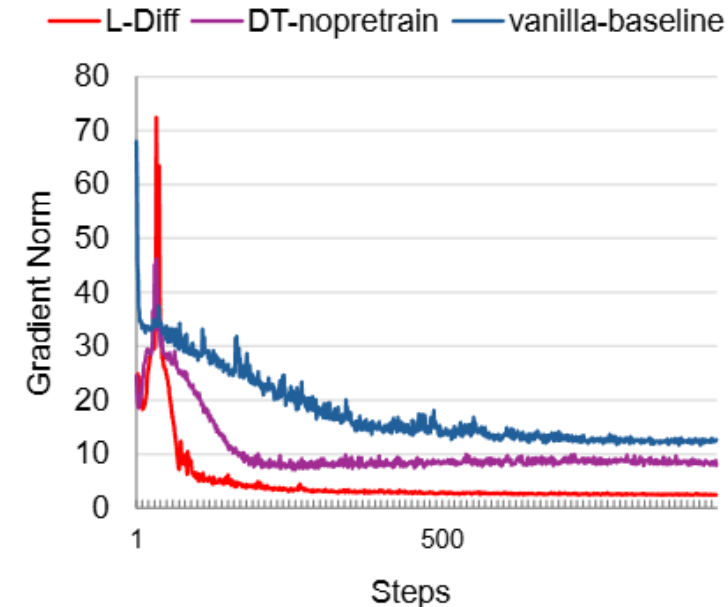


Figure 3: Gradient norm of the first 1000 steps during the pretraining.

# Experimental Results: Fine Tuning

Category	Specification
GPU	NVIDIA RTX A6000
Parameter Size	159M
Pretraining Dataset	WikiText-2
Training Precision	BFloat16
Max Sequence Length	2048

Table 1: Experimental Environments

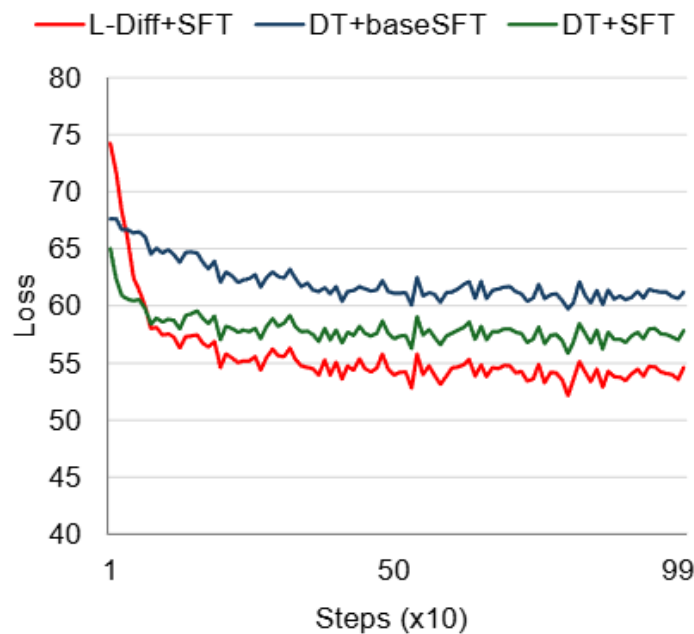


Figure 4: Loss of the first 1000 steps during the supervised fine tuning.

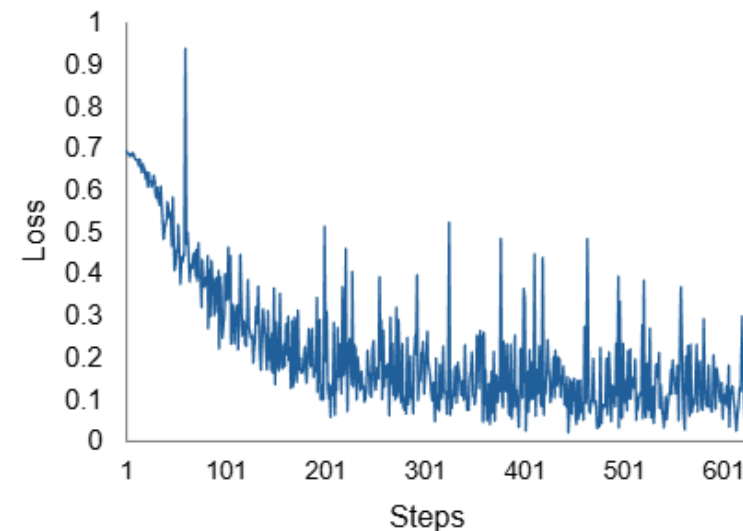


Figure 5: Loss of the first 625 steps during the direct preference optimization.

# Conclusion & Future Work

## We presented L-Diff, contributing with

- Fast & efficient Differential Transformer(DT) implementation using Flashattention kernel
- Optimized pretraining method including tensor alignment and learning rate scheduler
- 2-stage fine tuning pipeline using optimized SFT and DPO

We achieved **30%** lower early-stage loss and almost **28%** training time compared to the vanilla transformer baseline

## For the future work,

- Extend L-Diff to models exceeding 10B parameters
- Train & evaluate with large-scale datasets

# Thank You

## Q&A

### **L-Diff: Fast and Efficient Pretraining–Finetuning Framework for Differential Transformer-based Language Models**

**Yujeong Son**

Department of Electrical Engineering,  
Ulsan National Institute of Science and Technology (UNIST)  
syj4739@unist.ac.kr

**Eunseok Cho**

Artificial Intelligence Graduate School,  
Ulsan National Institute of Science and Technology (UNIST)  
eric754@unist.ac.kr