RevMUX: Data Multiplexing with Reversible Adapters for Efficient LLM Batch Inference

Paper Code

Yige Xu, Xu Guo, Zhiwei Zeng, Chunyan Miao
Nanyang Technological University, Singapore
{yige002,xu008}@e.ntu.edu.sg, {zhiwei.zeng,ascymiao}@ntu.edu.sg

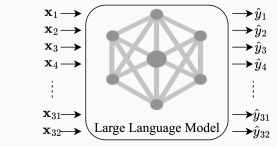
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Background

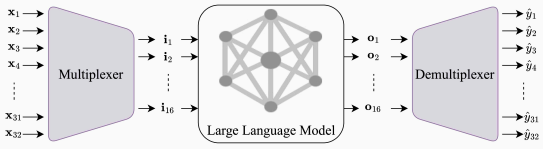
The expansion of Large Language Models (LLMs) has driven breakthroughs in Natural Language Processing (NLP) but raised concerns about inference efficiency, particularly latency, memory usage, and throughput.

This paper addresses the need for **high throughput** through **data multiplexing**, handling batches of concurrent queries while maintaining satisfactory downstream performance.

Mini-batch processing with Single-Input Single-Output (SISO)



Multi-input multi-output (MIMO) with data multiplexing and demultiplexing



Challenges & Motivations

The Multi-input-multi-output (MIMO) approach combines multiple inputs into a single one, enabling more efficient inference through a shared forward pass.

Challenges with Existing MIMO Models

Require **retraining** the LLM backbone along with the multiplexer and demultiplexer, which is impractical for increasingly large LLMs

Without retraining, the fixed LLMs **struggle to differentiate individual instances** within the consolidated inputs

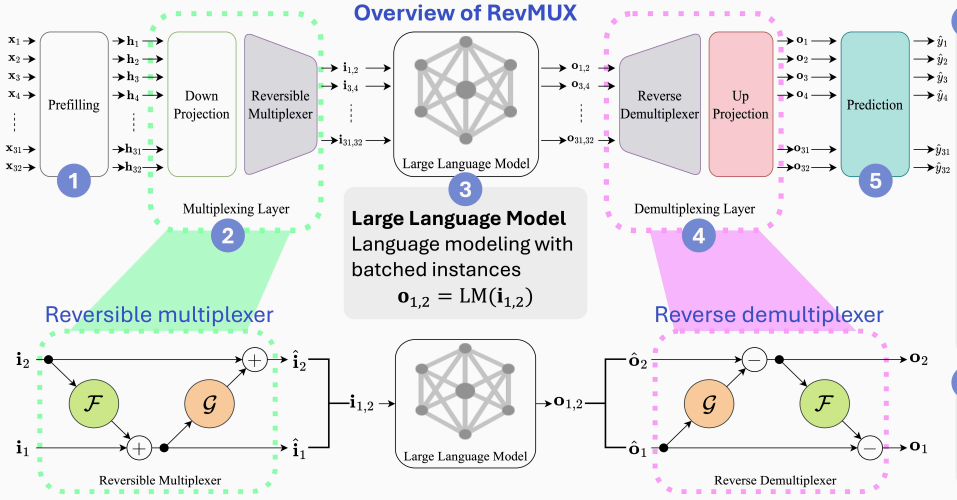
Our Approach

Fix the language model and tunes the adapters only

Design a reversible adapter to mix the instances and perform a reverse operation to reconstruct the individual inputs

Methodology

Overview of RevMUX



1 Prefilling

Prefilling for representation alignment

$$\mathbf{h} = \text{prefilling}(\mathbf{X})$$

2 Multiplexing Layer

Down projection $\mathbf{i} = f_{\text{down}}(\mathbf{h})$ for dimensional reduction: $\mathbb{R}^d \to \mathbb{R}^{d/N}$

Reversible multiplexer $N \times \mathbb{R}^{d/N} \to 1 \times \mathbb{R}^d$ for mixing instances

$$\hat{\mathbf{i}}_1 = \mathbf{i}_1 + \mathcal{F}(\mathbf{i}_2)$$
$$\hat{\mathbf{i}}_2 = \mathbf{i}_2 + \mathcal{G}(\hat{\mathbf{i}}_1)$$
$$\mathbf{i}_{1,2} = \text{concat}[\hat{\mathbf{i}}_1, \hat{\mathbf{i}}_2]$$
$$\mathcal{F}: \mathbb{R}^{d/N} \mapsto \mathbb{R}^{d/N}$$
$$\mathcal{G}: \mathbb{R}^{d/N} \mapsto \mathbb{R}^{d/N}$$

3 Large Language Model

Language modeling with batched instances

$$\mathbf{o}_{1,2} = \text{LM}(\mathbf{i}_{1,2})$$

4 Demultiplexing Layer

Reverse Demultiplexer $1 \times \mathbb{R}^d \to N \times \mathbb{R}^{d/N}$ for individual representation reconstruction

$$[\hat{\mathbf{o}}_1, \hat{\mathbf{o}}_2] = \mathbf{o}_{1,2}$$
$$\mathbf{o}_2 = \hat{\mathbf{o}}_2 - \mathcal{G}(\hat{\mathbf{o}}_1)$$
$$\mathbf{o}_1 = \hat{\mathbf{o}}_1 - \mathcal{F}(\mathbf{o}_2)$$

Up projection $f_{\text{up}}: \mathbb{R}^{d/N} \to \mathbb{R}^d$ for dimensional reconstruction

5 Prediction

$\mathcal{V}: \mathbb{R}^d \to \mathbb{R}^C$ for model output

Training Objectives

Cross-entropy Loss: from gold label for classification

$$\mathcal{L}_{\text{ce}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log(\hat{y}_{i,c})$$

InfoNCE Loss: encourage the demultiplexed representations to be close to those from the standard single-input single-output forward propagation

$$\mathcal{L}_{\text{info}} = -\frac{1}{N}\sum_{k=1}^{N}\mathbb{E}\left[\log\frac{\exp(\bar{\mathbf{o}}_k \cdot \mathbf{o}_k)}{\sum_{j=1}^{N}\exp(\bar{\mathbf{o}}_k \cdot \mathbf{o}_j)}\right]$$

Experiments & Results

Model Performance

- No fine-tune scenario is significantly more challenging
- RevMUX without fine-tune obtains comparable performance to fine-tuned MIMO baselines
- Fine-tuned RevMUX retains better performance than RevMUX without fine-tuned

		Model	N	↗	Tuned	Params	SST-2	MRPC	RTE	QNLI	Avg. Score
SISO	Backbones	$\text{BERT}_{\text{BASE}}$ (Devlin et al., 2019)	1	-	🔥	110M	92.20	87.01	62.96	90.55	83.18
		$\text{MUX-BERT}_{\text{BASE}}$ (Murahari et al., 2023)	1	100%	🔥	112M	91.74	87.75	63.18	90.54	83.30
MIMO	Baselines	DataMUX (Murahari et al., 2022)	2	180%	🔥	166M	90.50	85.05	60.87	88.39	81.20
		$\text{MUX-BERT}_{\text{BASE}}$ (Murahari et al., 2023)	2	201%	🔥	112M	90.62	83.77	58.19	88.17	80.19
	Ours	Vanilla Adapters	2	156%	❄	16.53M	90.42	84.78	60.06	88.19	80.86
		Only Multiplexer Reversible	2	161%	❄	20.07M	90.65	84.60	60.41	88.14	80.95
		RevMUX (❄)	2	154%	❄	9.45M	90.85	85.06	60.72	88.25	81.22
		RevMUX (🔥)	2	154%	🔥	120M	**91.21**	**85.78**	**61.41**	**88.72**	**81.78**

Scalability to Larger N

- RevMUX outperforms MUX-BERT-Base when N=2
- RevMUX maintains comparable or superior performance with larger N values

Model	N	Tuned	SST-2	MRPC	RTE	QNLI	Avg. Score
$\text{MUX-BERT}_{\text{BASE}}$	1	🔥	91.74	87.75	63.18	90.54	83.30
RevMUX	2	❄	90.85	85.06	60.72	88.25	81.22
$\text{MUX-BERT}_{\text{BASE}}$	2	🔥	90.62	83.77	58.19	88.17	80.19
RevMUX	4	❄	90.28	82.57	59.46	86.48	79.70
$\text{MUX-BERT}_{\text{BASE}}$	5	🔥	86.88	80.10	59.13	85.58	77.92
RevMUX	8	❄	88.30	78.97	58.66	85.17	77.78
$\text{MUX-BERT}_{\text{BASE}}$	10	🔥	83.44	78.63	58.27	82.08	75.61
RevMUX	16	❄	85.50	75.17	58.13	84.08	75.72

Module Ablation

- Vanilla Adapters < Only Multiplexer Reversible < RevMUX, demonstrates that both reversible multiplexer and reverse demultiplexer are effective

Backbones	Params.	Model	SST-2	MRPC	RTE	QNLI	Avg. Score
$\text{BERT}_{\text{BASE}}$	110M	Vanilla Adapters	90.42	84.78	60.06	88.19	80.86
		Only Multiplexer Reversible	90.65	84.60	60.41	88.14	80.95
		RevMUX	**90.85**	**85.06**	**60.72**	**88.25**	**81.22**
T5_{Small}	60M	Vanilla Adapters	89.00	81.72	57.22	85.36	78.33
		Only Multiplexer Reversible	89.04	82.30	57.51	85.44	78.57
		RevMUX	**89.14**	**82.45**	**60.22**	**85.63**	**79.36**
T5_{Base}	220M	Vanilla Adapters	92.36	82.94	63.28	87.58	81.54
		Only Multiplexer Reversible	92.54	83.19	64.01	88.14	81.98
		RevMUX	**92.70**	**83.80**	**64.73**	**88.65**	**82.47**
T5_{Large}	770M	Vanilla Adapters	92.58	83.16	64.22	88.42	82.10
		Only Multiplexer Reversible	92.67	83.46	64.43	88.56	82.28
		RevMUX	**92.81**	**83.86**	**65.01**	**88.89**	**82.64**
LLaMA3-8B	8B	Vanilla Adapters	94.01	80.96	82.72	85.99	85.92
		Only Multiplexer Reversible	94.09	81.08	82.82	86.24	86.06
		RevMUX	**94.38**	**81.30**	**83.18**	**86.53**	**86.35**

Scalability to Larger Models

- RevMUX is scalable to billion-scale decoder-only LLMs
- RevMUX retains performance while improving efficiency
- The reversible design remains effective on larger-scale LLMs