Fast and Low-Cost Genomic Foundation Models via Outlier Removal

Haozheng Luo*† Chenghao Qiu*♮ Maojiang Su† Zhihan Zhou† Zoe Mehta† Guo Ye† Jerry Yao-Chieh Hu† Han Liu†

†Northwestern University, Evanston, IL 60208 USA; ♮Tianjin University, Tianjin 300350, China

Northwestern University
天津大学 Tianjin University

Summary: Outlier-Free Genomic Foundation Model

- We introduce a new **outlier-free** Genomic Foundation Model (GFM) architecture called **GERM**, featuring superior **post-quantization** and **low-rank adaptation** performance.
- GERM retains and improves the desirable properties of GFMs in quantization and low-rank adaptation
- All DNABERT fine-tuning tasks finish in **only 5 minutes** on a single NVIDIA GeForce RTX 2080 Ti GPU.
- Achieves average performance improvements of 37.98% in finetuning and 64.34% in quantization, with 92.14% lower average kurtosis and 82.77% lower mmaximum infinity norm $|\mathbf{x}|_\infty$ values.

Background: Outliers in Attention Heads & GFMs

Attention Outlier [Hu et al., 2024] identify attention outliers where certain tokens induce a "wide range" in $QK^\top$, referred to as *no-op outliers*. In GFMs, tokens or activations that disproportionately influence the attention mechanism with:

- Tokens with little or no meaningful information receive disproportionately high attention weights.
- Recurring nucleotide patterns are overemphasized by $\mathrm{Softmax}$.

Genomic Foundation Models Large-scale pretrained models designed for modeling and analysing genomic sequences.

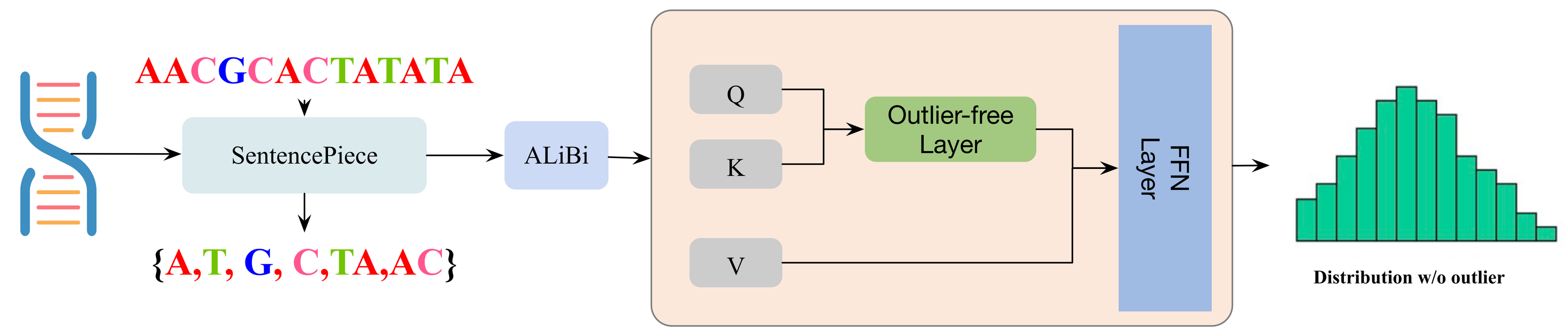
- Trained on massive genomic datasets
- **Classification models**: e.g., `DNABERT-2`, `Nucleotide Transformer (NT)`, `HyenaDNA`
- **Generative models**: e.g., `Evo`, `GenomeOcean`
- Larger GFMs, especially generative models, require substantial computational resources for deployment and fine-tuning.

GERM

GERM is a GFM architecture with an outlier-free layer that enhances post-quantization and low-rank adaptation performance.

- We propose a new GFM architecture GERM by replacing the $\mathrm{Softmax}$ in the attention mechanism with $\mathrm{Softmax}_1$ to achieve the Quantization Robustness and Fast Low-rank Adaptation.

$$\mathrm{Softmax}_1(S) := \frac{\exp(S)}{1 + \sum_{i=1}^{L} \exp(S_i)},$$



- The original OutEffHop method requires training from scratch; we propose a trade-off variant, **GERM-T**, to achieve sub-optimal performance with small-step continual learning.

Experimental Studies: Significant Reduce Outlier

GERM significantly reduces $\|\mathbf{x}\|_\infty$ compared to vanilla attention and enhances post-quantization performance.

Model	#Bits	Quantization Method	MCC (↑)	Delta MCC (↓)	Avg Performance Drop (↓)	Avg. Kurtosis (↓)	Max inf. norm (↓)
Official	16W/16A	-	66.11	-	-	39.68	53.61
DNABERT-2	16W/16A	-	59.11	7.00	-	270.90	61.64
	8W/8A	-	33.60±0.41	32.51	43.81%		
	8W/8A	SmoothQuant	36.51±0.02	45.37	38.63%		
	6W/6A	SmoothQuant	20.74±0.04	45.37	66.18%		
	4W/4A	SmoothQuant	-1.03±0.06	67.06	101.24%		
	8W/8A	Outlier	25.26±0.02	40.85	57.60%		
	6W/6A	Outlier	27.84±0.28	38.27	52.71%		
	8W/8A	OmniQuant	49.92±0.05	16.19	15.76%		
	6W/6A	OmniQuant	48.47±0.14	17.64	18.61%		
	4W/4A	OmniQuant	2.94±0.19	63.17	94.78%		
GERM	16W/16A	-	59.73	6.38	-	**21.29**	**10.62**
	8W/8A	-	57.30±0.08	8.81	**3.77%**		
	8W/8A	SmoothQuant	56.65±0.15	9.46	4.82%		
	6W/6A	SmoothQuant	56.48±0.07	9.63	**5.45%**		
	4W/4A	SmoothQuant	20.05±0.00	46.06	**69.44%**		
	8W/8A	Outlier	45.87±0.08	20.24	**25.23%**		
	6W/6A	Outlier	40.57±0.56	25.54	36.27%		
	8W/8A	OmniQuant	55.99±0.09	10.12	5.95%		
	6W/6A	OmniQuant	55.70±0.03	10.41	**6.41%**		
	4W/4A	OmniQuant	49.42±0.00	16.69	**17.17%**		
GERM-T	16W/16A	-	59.30	6.81	-	251.40	28.49
	8W/8A	-	38.38±0.15	27.73	35.27%		
	8W/8A	SmoothQuant	57.52±0.00	8.59	**3.01%**		
	6W/6A	SmoothQuant	30.34±0.04	35.77	48.83%		
	4W/4A	SmoothQuant	0.22±0.00	65.89	99.63%		
	8W/8A	Outlier	42.57±0.05	23.54	28.31%		
	6W/6A	Outlier	46.02±0.06	20.06	**22.34%**		
	8W/8A	OmniQuant	56.80±0.12	9.31	**4.21%**		
	6W/6A	OmniQuant	55.41±0.00	10.71	6.57%		
	4W/4A	OmniQuant	3.86±0.00	62.25	93.49%		

Outlier Efficiency of Low-Rank Adaptation

- GERM reduces 92+% in average kurtosis and 82+% in the maximum infinity norm of model's outputs in DNABERT-2.
- GERM achieves a 37.98% improvement in model fine-tuning performance.

Models	Low-Rank Adaptation Method	MCC (↑)	Delta MCC different (↓)	Avg Performance Drop (↓)	Avg. kurtosis(↓)	Max inf. norm(↓)
DNABERT-2	Full	59.11	7.00	-	270.90	61.41
	LoRA	50.91±1.67	15.2	13.87%	-	219.20
	QLoRA	50.65±0.13	15.46	14.31%	292.85	53.91
	LoftQ	50.76±0.06	15.31	14.05%	299.18	54.18
GERM	Full	59.73	6.38	-	**21.29**	**10.62**
	LoRA	57.27±0.70	8.84	**4.12%**	-	**19.41**
	QLoRA	53.16±0.21	12.95	**10.99%**	**34.29**	**27.27**
	LoftQ	53.11±0.08	13.00	**11.08%**	**33.02**	**27.41**
GERM-T	Full	59.30	6.81	-	251.40	28.49
	LoRA	55.60±0.28	10.51	6.23%	-	140.86
	QLoRA	51.05±0.07	15.06	13.90%	287.95	53.92
	LoftQ	51.20±0.13	14.91	13.65%	286.16	53.35

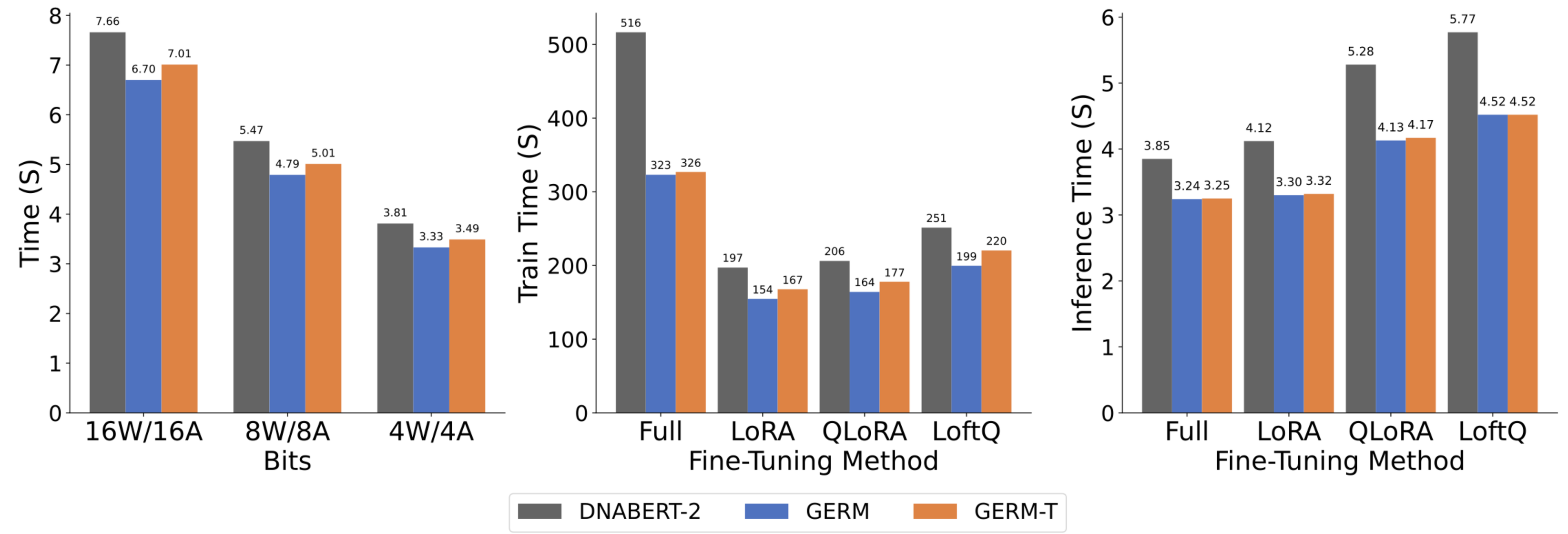
We also observe a significant performance improvement on the Nucleotide Transformer 2.5B model; please refer to our paper for details.

- GERM achieves a 50.83% in PTQ performance and 66.02% in Low-rank adaptation performance.
- GERM-T achieves a 36.73% in PTQ performance and 34.56% in Low-rank adaptation performance.

Experimental Studies: Efficient GFM with GERM.

GERM reduces both fine-tuning and inference time

- GERM reduces fine-tuning time by 34.85% and improves inference latency by 24.79%.
- GERM-T reduces fine-tuning time by 26.68% and improves inference latency by 24.21%.



Performance Improve on CPU-only Environment

- GERM reduces fine-tuning time by 37.32% and improves inference latency by 33.95%.
- GERM-T reduces fine-tuning time by 21.63% and improves inference latency by 30.25%.

Method	Fine-Tuning Method	MCC (↑)	Time (sec.) Train	Time (sec.) Inference
DNABERT-2	LoRA	50.91	808.23	29.66
GERM	LoRA	**57.27**	**618.68**	**23.10**
GERM-T	LoRA	55.60	674.40	23.57
DNABERT-2	QLoRA	50.65	516.04	63.17
GERM	QLoRA	**53.16**	**358.34**	**45.28**
GERM-T	QLoRA	51.50	418.13	46.91

More Experiments: Influence of Continual Learning.

GERM-T shows the smallest performance drop during quantization and low-rank adaptation compared to other continual learning steps.

Method	Fine-Tuning Method	MCC (↑)	Avg Performance Drop (↓)
DNABERT-2	Full	59.11	-
GERM	Full	59.73	-
Out20k	Full	59.21	-
GERM-T	Full	59.30	-
Out100k	Full	60.56	-
DNABERT-2	LoRA	50.91	13.87%
GERM	LoRA	56.78	**4.94%**
Out20k	LoRA	54.75	7.53%
GERM-T	LoRA	55.60	6.24%
Out100k	LoRA	56.61	6.52%
DNABERT-2	QLoRA	50.65	14.31%
GERM	QLoRA	53.16	**11.00%**
Out20k	QLoRA	50.61	14.52%
GERM-T	QLoRA	51.05	13.91%
Out100k	QLoRA	51.24	15.39%
DNABERT-2	LoftQ	50.76	14.13%
GERM	LoftQ	53.11	**11.08%**
Out20k	LoftQ	50.94	13.97%
GERM-T	LoftQ	51.20	13.66%
Out100k	LoftQ	50.77	16.17%