Understanding Linear Probing then Fine-tuning Language Models from NTK Perspective

Akiyoshi Tomihari, Issei Sato

The University of Tokyo

Linear Probing then Fine-tuning (LP-FT)

- LP-FT is a fine-tuning method [Kumar et al., 2022]
  - 1st Linear probing (LP), 2nd Fine-tuning (FT)
- FT starts with the optimized linear layer (classifier).

Changes to pre-trained features are minimized.

- Problem: Existing analyses focus on two-layer linear models.

LP-FT from NTK perspective

- Use *neural tangent kernel (NTK) theory* in fine-tuning. [Malladi et al., 2023]
- The classifier norm affects the NTK and changes in features.

Feature extractor

$$\phi^{FT}(\boldsymbol{x}) - \phi_0(\boldsymbol{x}) = \eta \sum_{i=1}^{N} \underbrace{\Theta^{\phi}(\boldsymbol{x}, \boldsymbol{x}_i)}_{\text{Feature extractor NTK } O(1)} \boldsymbol{V}_0^{\top} \boldsymbol{\delta}_i$$

Model

$$\boldsymbol{f}^{FT}(\boldsymbol{x}) - \boldsymbol{f}_0(\boldsymbol{x}) = \eta \sum_{i=1}^{N} \Big(\underbrace{\boldsymbol{P}(\boldsymbol{x}, \boldsymbol{x}_i)}_{\text{Sparse}} + \underbrace{\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{x}_i)}_{\text{Dense}} \Big) \boldsymbol{\delta}_i$$

Pre-train-effective $O(1)$

FT-effective $O(|V_0|^2)$

V_0: Classifier

Increase in classifier norms

- The derivative of empirical risk with respect to the norm of a classifier row vector becomes negative.

➡ **Classifier norms increase during training.**

$$\underbrace{\frac{\partial L(\boldsymbol{f})}{\partial \|\boldsymbol{v}_k\|}}_{\leq 0} = \sum_{i=1}^{N} \left(\sum_{k \neq y_i} [\boldsymbol{\sigma}_{\mathrm{SM}}(\boldsymbol{f}(\boldsymbol{x_i}))]_k \|\boldsymbol{\phi}(\boldsymbol{x_i})\| \underbrace{\cos \tau_{ki}}_{\leq 0} - \sum_{k=y_i} (1 - [\boldsymbol{\sigma}_{\mathrm{SM}}(\boldsymbol{f}(\boldsymbol{x_i}))]_{y_i}) \|\boldsymbol{\phi}(\boldsymbol{x_i})\| \underbrace{\cos \tau_{y_i i}}_{\geq 0} \right).$$

Small feature changes in LP-FT

- Changes in feature are smaller in LP-FT than FT.
- LP-FT mitigates feature distortion in language models.

Features (F) and classifier norms (C) analysis: cosine similality (CS), difference (Diff), and Fisher discriminant ratio (FDR).

Method	CB				RTE			
	CS(F)	Diff(F)	FDR(F)	Norm(C)	CS(F)	Diff(F)	FDR(F)	Norm(C)
Pre-trained	0.997	–	8.14×10^4	9.51×10^{-1}	0.996	–	8.59×10^1	7.76×10^{-1}
LP	0.997	–	8.14×10^4	2.48×10^1	0.996	–	8.59×10^1	3.10×10^1
FT	0.336	2.21×10^1	7.39×10^8	9.60×10^{-1}	0.260	2.16×10^1	1.42×10^4	7.84×10^{-1}
LoRA	0.499	1.92×10^1	8.91×10^6	1.43×10^0	0.759	1.06×10^1	2.97×10^3	1.21×10^0
LP-FT	0.804	1.20×10^1	6.47×10^6	2.48×10^1	0.942	4.70×10^0	1.57×10^2	3.10×10^1
LP-LoRA	0.837	9.08×10^0	2.10×10^6	2.49×10^1	0.924	4.63×10^0	2.06×10^1	3.10×10^1

Small feature changes in LP-FT

- Changes in feature are smaller in LP-FT than FT.
- LP-FT mitigates feature distortion in language models.

Features (F) and classifier norms (C) analysis: cosine similality (CS), difference (Diff), and Fisher discriminant ratio (FDR).

Method	CB				RTE			
	CS(F)	Diff(F)	FDR(F)	Norm(C)	CS(F)	Diff(F)	FDR(F)	Norm(C)
Pre-trained	0.997	–	8.14×10^{4}	9.51×10^{-1}	0.996	–	8.59×10^{1}	7.76×10^{-1}
LP	0.997	–	8.14×10^{4}	2.48×10^{1}	0.996	–	8.59×10^{1}	3.10×10^{1}
FT	0.336	2.21×10^{1}	7.39×10^{8}	9.60×10^{-1}	0.260	2.16×10^{1}	1.42×10^{4}	7.84×10^{-1}
LoRA	0.499	1.92×10^{1}	8.91×10^{6}	1.43×10^{0}	0.759	1.06×10^{1}	2.97×10^{3}	1.21×10^{0}
LP-FT	0.804	1.20×10^{1}	6.47×10^{6}	2.48×10^{1}	0.942	4.70×10^{0}	1.57×10^{2}	3.10×10^{1}
LP-LoRA	0.837	9.08×10^{0}	2.10×10^{6}	2.49×10^{1}	0.924	4.63×10^{0}	2.06×10^{1}	3.10×10^{1}

Small feature changes and large classifier norms in LP-FT.

Small feature changes in LP-FT

- Changes in feature are smaller in LP-FT than FT.
- LP-FT mitigates feature distortion in language models.

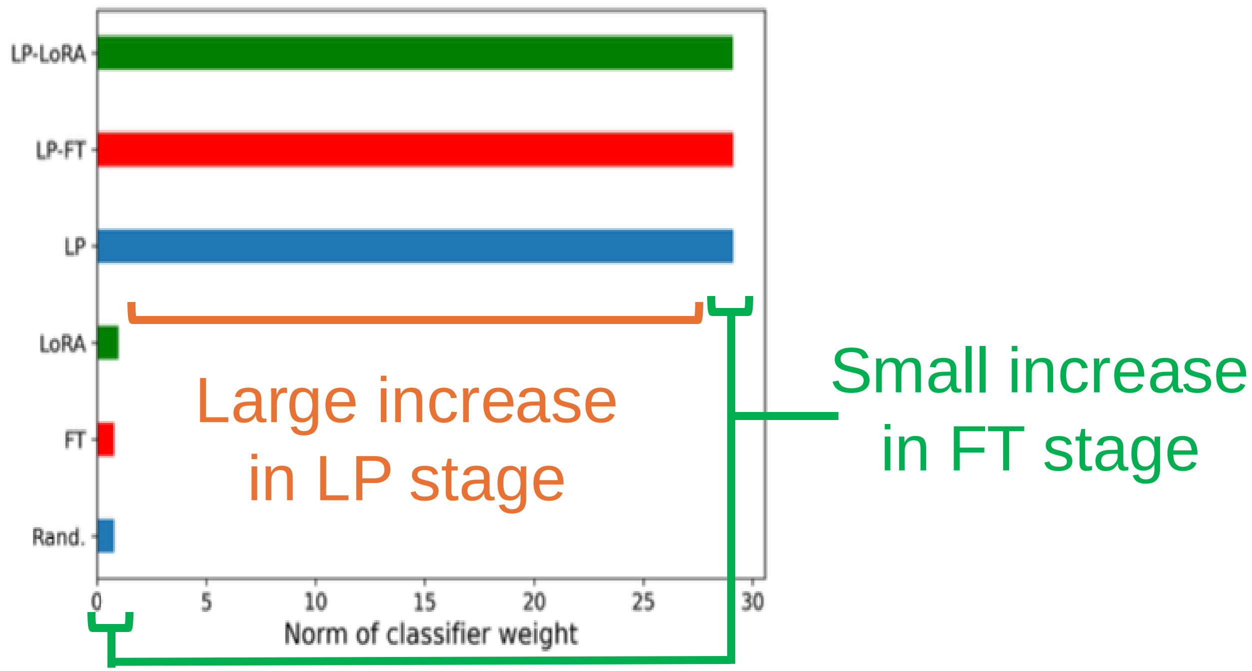
Features (F) and classifier norms (C) analysis: cosine similality (CS), difference (Diff), and Fisher discriminant ratio (FDR).

Method	CB				RTE			
	CS(F)	Diff(F)	FDR(F)	Norm(C)	CS(F)	Diff(F)	FDR(F)	Norm(C)
Pre-trained	0.997	–	8.14×10^4	9.51×10^{-1}	0.996	–	8.59×10^1	7.76×10^{-1}
LP	0.997	–	8.14×10^4	2.48×10^1	0.996	–	8.59×10^1	3.10×10^1
FT	0.336	2.21×10^1	7.39×10^8	9.60×10^{-1}	0.260	2.16×10^1	1.42×10^4	7.84×10^{-1}
LoRA	0.499	1.92×10^1	8.91×10^6	1.43×10^0	0.759	1.06×10^1	2.97×10^3	1.21×10^0
LP-FT	0.804	1.20×10^1	6.47×10^6	2.48×10^1	0.942	4.70×10^0	1.57×10^2	3.10×10^1
LP-LoRA	0.837	9.08×10^0	2.10×10^6	2.49×10^1	0.924	4.63×10^0	2.06×10^1	3.10×10^1

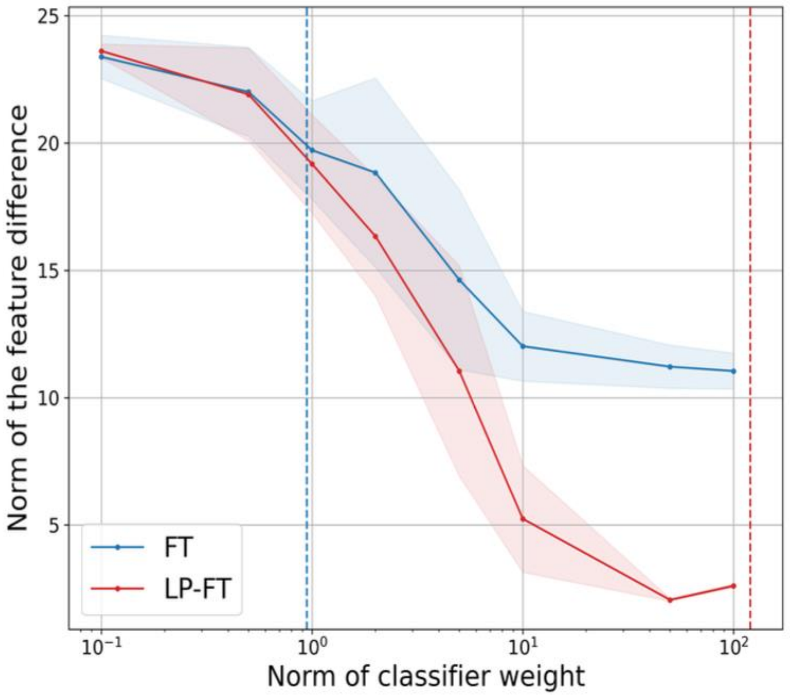
The characteristics of pretrained features (high CS and low FDR) are preserved in LP-FT.

Increase in classifier norms

- Classifier norms increase in LP stage.
- Increased classifier norms reduce changes in features.



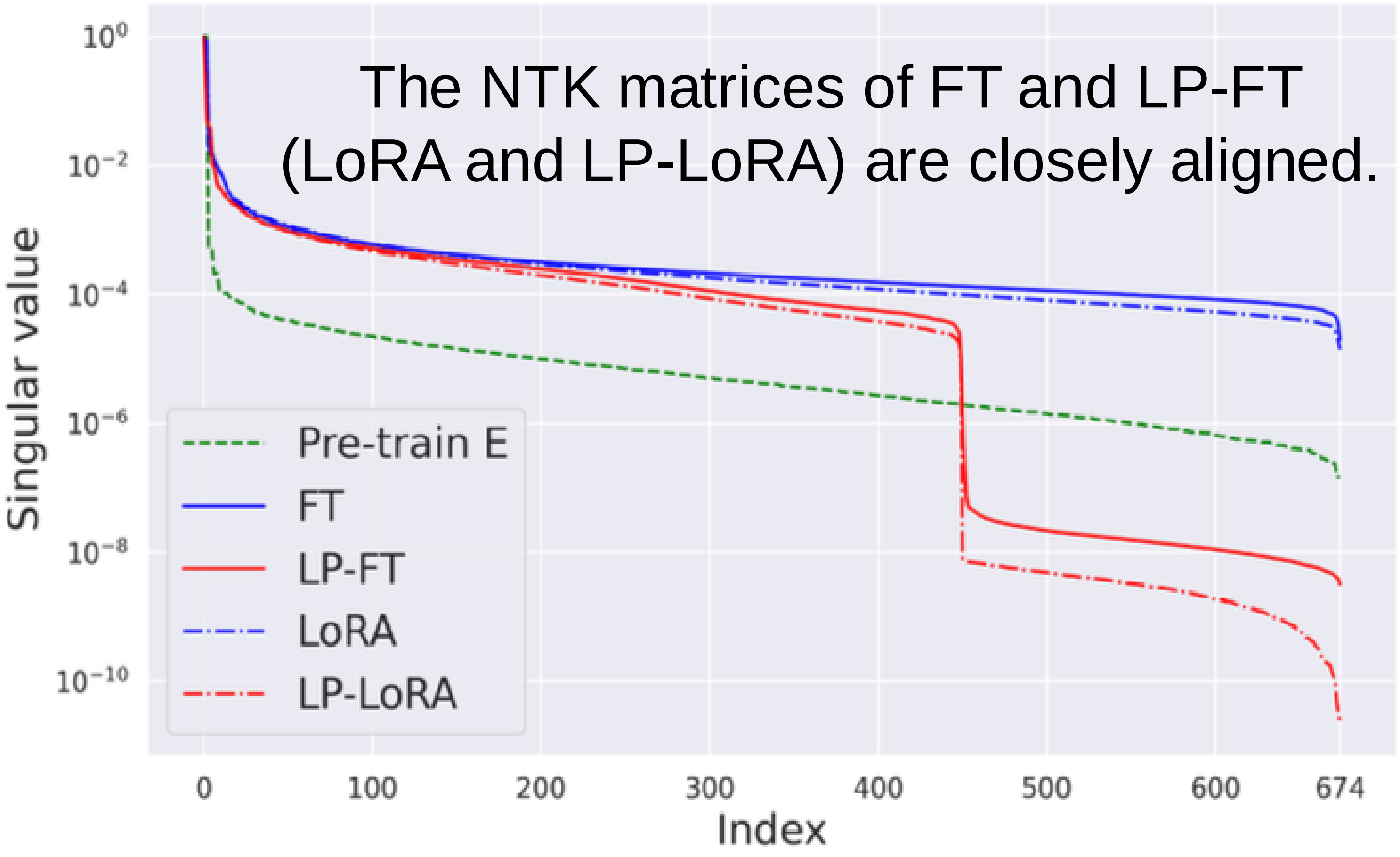
Classifier norms after training



Classifier norms and feature changes

NTK matrix of LoRA

- NTK martrices of FT and LoRA is similar.
- This similarity suggests that LoRA effectively approximates FT.



Singular value distribution of NTK matrices

Temperature scaling at test time

- Increased classifier norms can distort probability alignments, impact *model calibration*.
- *Temperature scaling* at test time can mitigates this effect.

Metric	Method	w/o TS	w/ TS	Imp.
ECE (%)	FT	21.16	5.13	16.03
	LP-FT	21.72	5.48	**16.24**
	LoRA	11.92	6.17	5.76
	LP-LoRA	18.14	5.72	12.42
MCE (%)	FT	53.11	25.87	27.24
	LP-FT	63.95	13.94	**50.01**
	LoRA	25.04	13.75	11.29
	LP-LoRA	40.46	18.82	21.63

$$\boldsymbol{f}(\boldsymbol{x})/T = \frac{\boldsymbol{V}}{T}\phi(\boldsymbol{x}) + \frac{\boldsymbol{b}}{T}$$

Temperature scaling effectively enhances the calibration of LP-FT.

Conclusion

- Analyze LP-FT from NTK perspective.
- Highlight the importance of the classifier norm during training.
- Observe a trend of increase in the classifier norm.
- Demonstrate LP-FT mitigates feature distortion in language models.
- Verify the effectiveness of LoRA and temperature scaling.

Paper link:
https://arxiv.org/abs/2405.16747