Once Read is Enough: Domain-specific Pretraining-free Language Models with Cluster-guided Sparse Experts for Long-tail Domain Knowledge

Fang Dong, Mengyi Chen, Jixian Zhou, Yubin Shi, Yixuan Chen, Mingzhi Dong, Yujiang Wang,

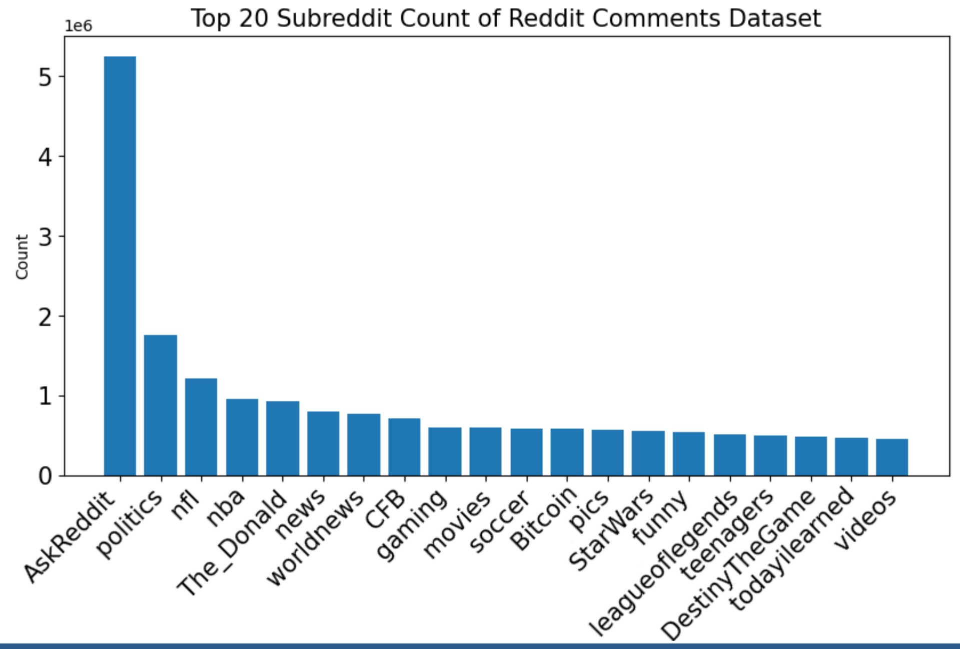
Dongsheng Li, Xiaochen Yang, Rui Zhu, Robert Dick, Qin Lv, Fan Yang, Tun Lu, Ning Gu, Li Shang

Introduction

- Pretrained models usually fail to expertise in downstream tasks requiring specialized domain knowledge.

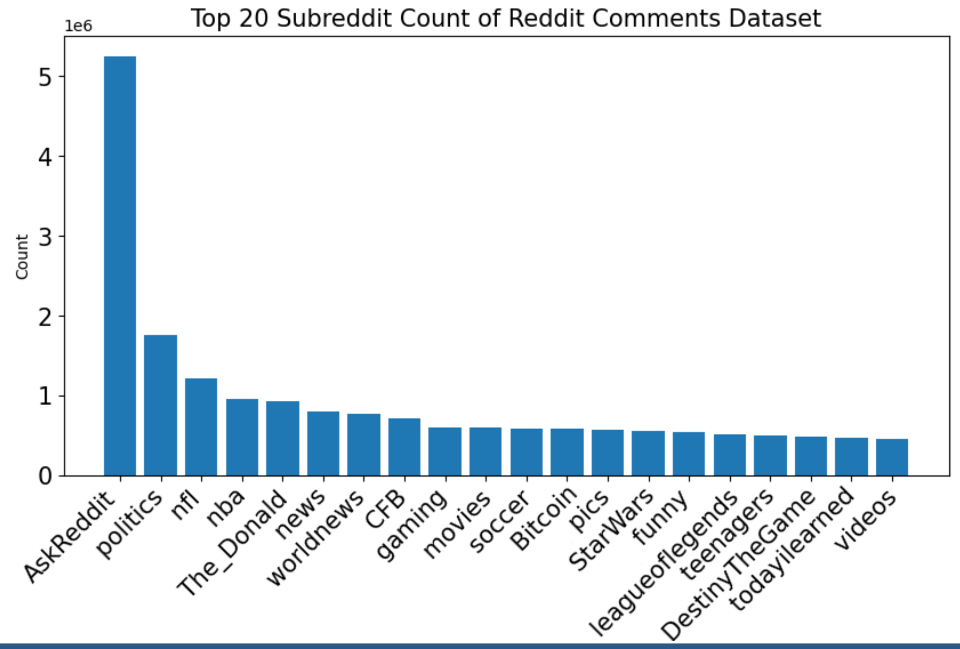
Introduction

- Pretrained models usually fail to expertise in downstream tasks requiring specialized domain knowledge.
- Domain-specific information appears significantly less frequently than general knowledge, or long-tail.



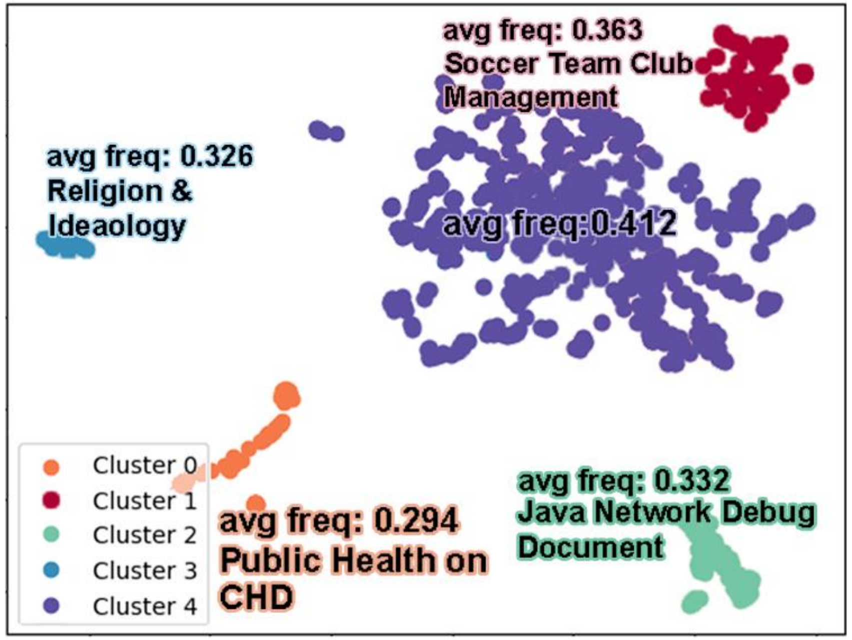
Analysis

- Data domains show a long-tail distribution.



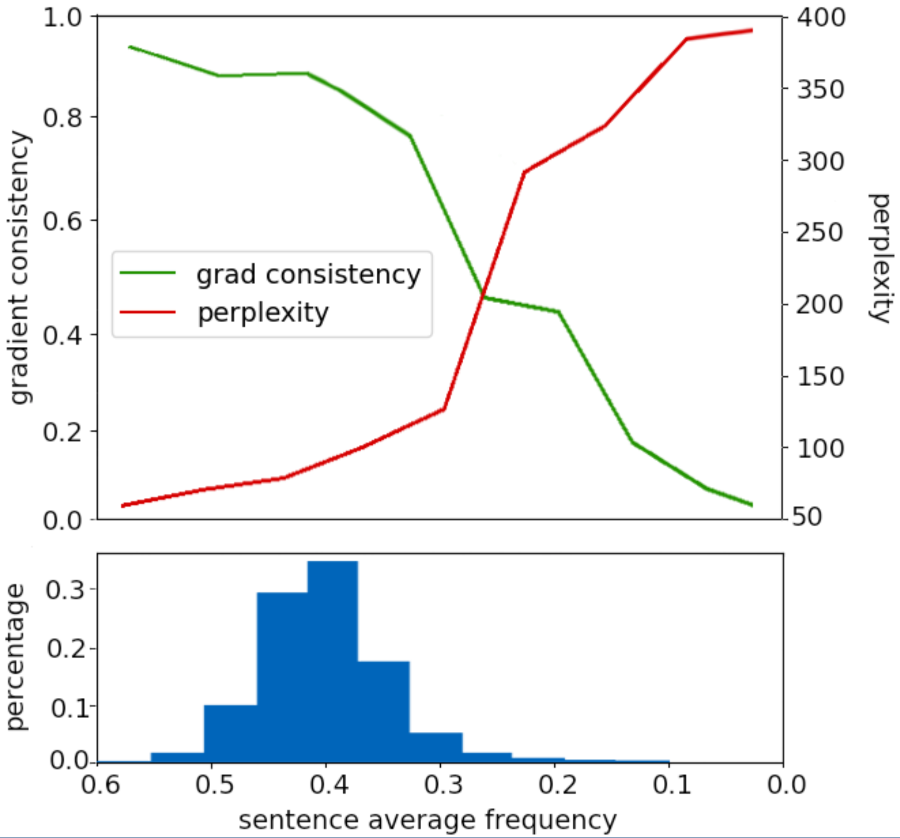
Analysis

- Data domains show a long-tail distribution.
- Lower frequency score for domain-specific data
  - Frequency score is defined as average token frequency in a sentence
  - Low-frequency score data are considered long-tail



Analysis

- Data domains show a long-tail distribution.
- Lower frequency score for domain-specific data
- Data with lower frequency show lower gradient consistency and higher perplexity



Analysis

- Data with lower frequency show lower gradient consistency and higher perplexity

Neural **T**angent **K**ernel: $\boldsymbol{\Theta}(\mathcal{X}, \mathcal{X}) = J_{\boldsymbol{\theta}}(\mathcal{X}) J_{\boldsymbol{\theta}}(\mathcal{X})^{\top}$, where Jacobian $J_{\boldsymbol{\theta}} = \nabla_{\boldsymbol{\theta}} f(\mathcal{X}; \boldsymbol{\theta})$

Analysis

- Data with lower frequency show lower gradient consistency and higher perplexity

Neural **T**angent **K**ernel: $\boldsymbol{\Theta}(\mathcal{X},\mathcal{X}) = J_{\boldsymbol{\theta}}(\mathcal{X})J_{\boldsymbol{\theta}}(\mathcal{X})^\top$, where Jacobian $J_{\boldsymbol{\theta}} = \nabla_{\boldsymbol{\theta}} f(\mathcal{X};\boldsymbol{\theta})$

PCA on NTK: $\boldsymbol{\Theta} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top = \sum_{i=1}^{n} \lambda_i \mathbf{u}_i \mathbf{u}_i^\top$, and $\mathbf{u}_{max}$ corresponds to the max eigenvalue

Analysis

- Data with lower frequency show lower gradient consistency and higher perplexity

Neural **T**angent **K**ernel: $\boldsymbol{\Theta}(\mathcal{X},\mathcal{X}) = J_{\boldsymbol{\theta}}(\mathcal{X})J_{\boldsymbol{\theta}}(\mathcal{X})^\top$, where Jacobian $J_{\boldsymbol{\theta}} = \nabla_{\boldsymbol{\theta}} f(\mathcal{X};\boldsymbol{\theta})$

PCA on NTK: $\boldsymbol{\Theta} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top = \sum_{i=1}^{n} \lambda_i \mathbf{u}_i \mathbf{u}_i^\top$, and $\mathbf{u}_{max}$ corresponds to the max eigenvalue

Primary Gradient Direction: $\mathbf{g}_{\boldsymbol{\theta}}(\mathcal{X}) = \mathbf{u}_{max} J_{\boldsymbol{\theta}}(\mathcal{X})$

Analysis

- Data with lower frequency show lower gradient consistency and higher perplexity

Neural **T**angent **K**ernel: $\boldsymbol{\Theta}(\mathcal{X},\mathcal{X}) = J_{\boldsymbol{\theta}}(\mathcal{X})J_{\boldsymbol{\theta}}(\mathcal{X})^{\top}$, where Jacobian $J_{\boldsymbol{\theta}} = \nabla_{\boldsymbol{\theta}} f(\mathcal{X};\boldsymbol{\theta})$

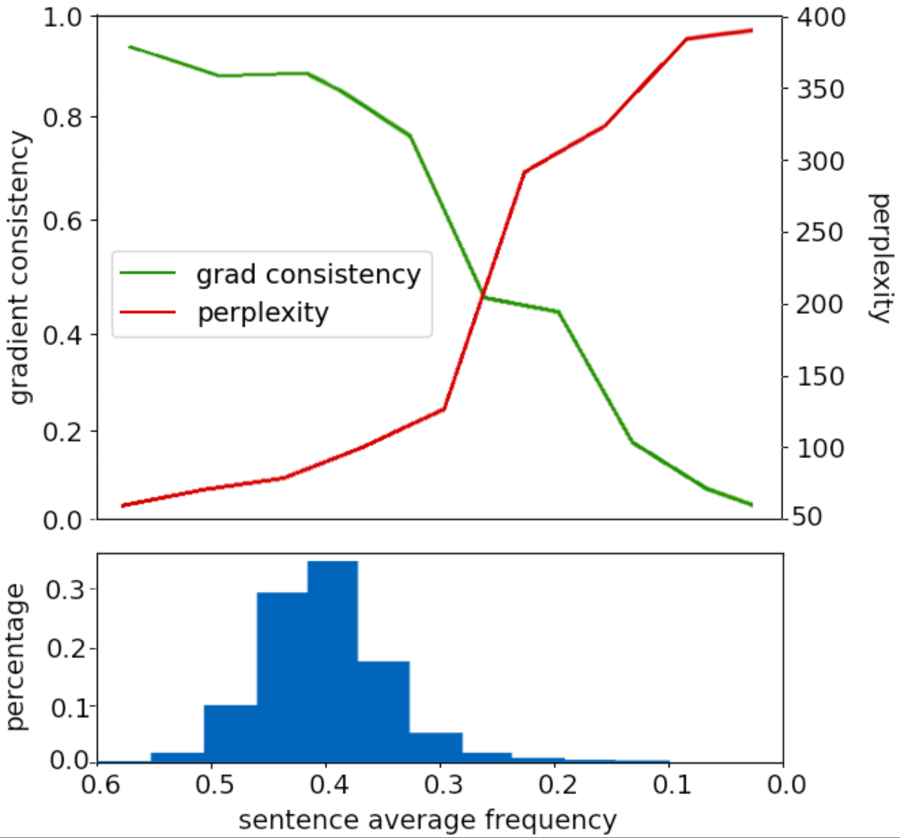
PCA on NTK: $\boldsymbol{\Theta} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\top} = \sum_{i=1}^{n} \lambda_i \mathbf{u}_i \mathbf{u}_i^{\top}$, and $\mathbf{u}_{max}$ corresponds to the max eigenvalue

Primary Gradient Direction: $\mathbf{g}_{\boldsymbol{\theta}}(\mathcal{X}) = \mathbf{u}_{max} J_{\boldsymbol{\theta}}(\mathcal{X})$

Gradient **C**onsistency: $GC_{\theta}(X') = \dfrac{\mathbf{g}_{\boldsymbol{\theta}}(\mathcal{X}) \cdot \mathbf{g}_{\boldsymbol{\theta}}(\mathcal{X}')}{\|\mathbf{g}_{\boldsymbol{\theta}}(\mathcal{X})\| \|\mathbf{g}_{\boldsymbol{\theta}}(\mathcal{X}')\|}$

Analysis

- Data domains show a long-tail distribution.
- Lower frequency score for domain-specific data
- Data with lower frequency show lower gradient consistency and higher perplexity



Method

- Cluster-guided Sparse Expert:
  - **Initialization**: train a baseline dense model devoid of any expert structure.

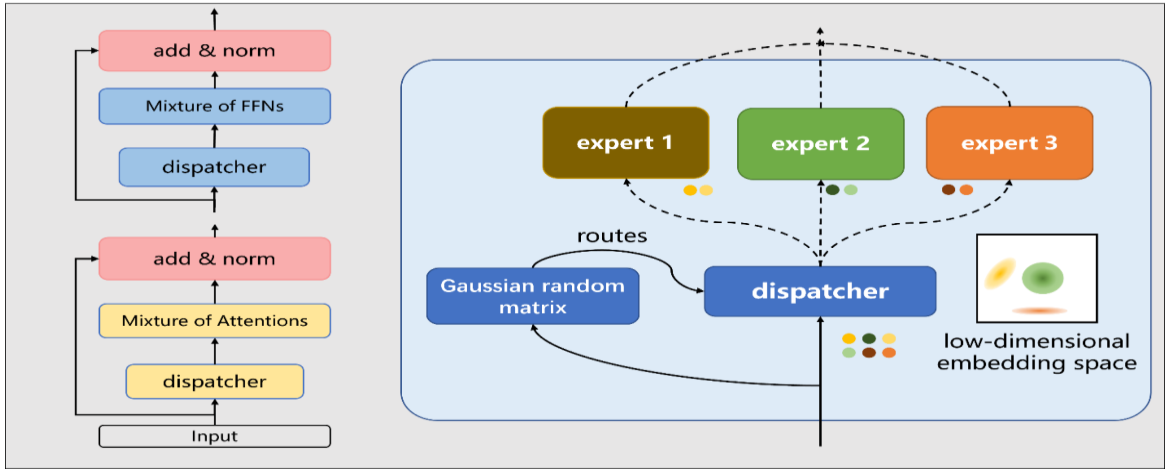
Method

- Cluster-guided Sparse Expert:
  - **Initialization**: train a baseline dense model devoid of any expert structure.
  - **Low-Dimension Clustering**: use a Gaussian random initialized matrix to project embeddings to a low-dimensional space, and perform clustering algorithm in all layers.

Method

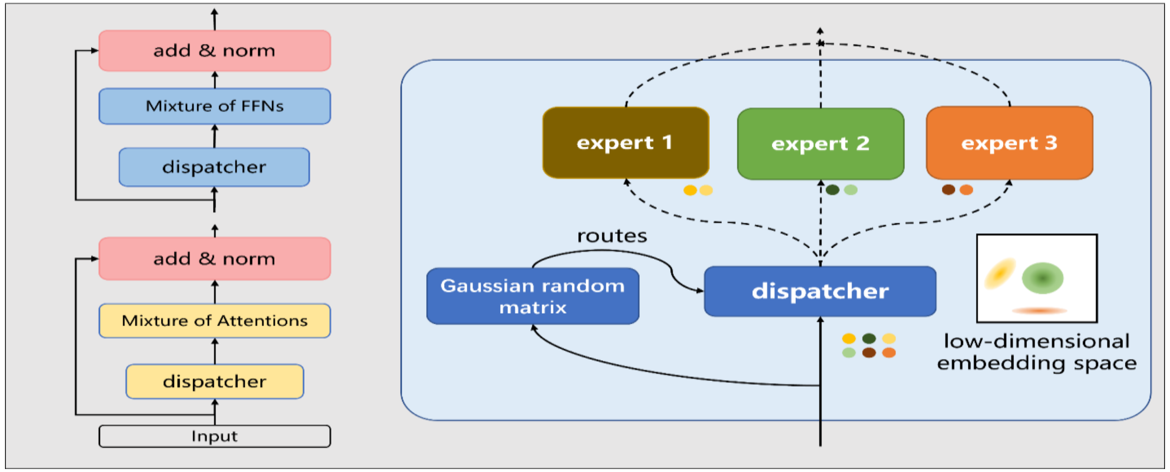
- Cluster-guided Sparse Expert:
  - **Initialization**: train a baseline dense model devoid of any expert structure.
  - **Low-Dimension Clustering**: use a Gaussian random initialized matrix to project embeddings to a low-dimensional space, and perform clustering algorithm in all layers.
  - **Select Layer**: introduce MoE on layers with larger cluster distance-radii ratio. Expert number is equal to the cluster number.

Method



- Cluster-guided Sparse Expert:
  - **Initialization**: train a baseline dense model devoid of any expert structure.
  - **Low-Dimension Clustering**: use a Gaussian random initialized matrix to project embeddings to a low-dimensional space, and perform clustering algorithm in all layers.
  - **Select Layer**: introduce MoE on layers with larger cluster distance-radii ratio. Expert number is equal to the cluster number.
  - **Dispatch**: dispatch new data to its nearest cluster $i = \arg\min_{j=1}^{n} ||v' - c_j||/r_j$

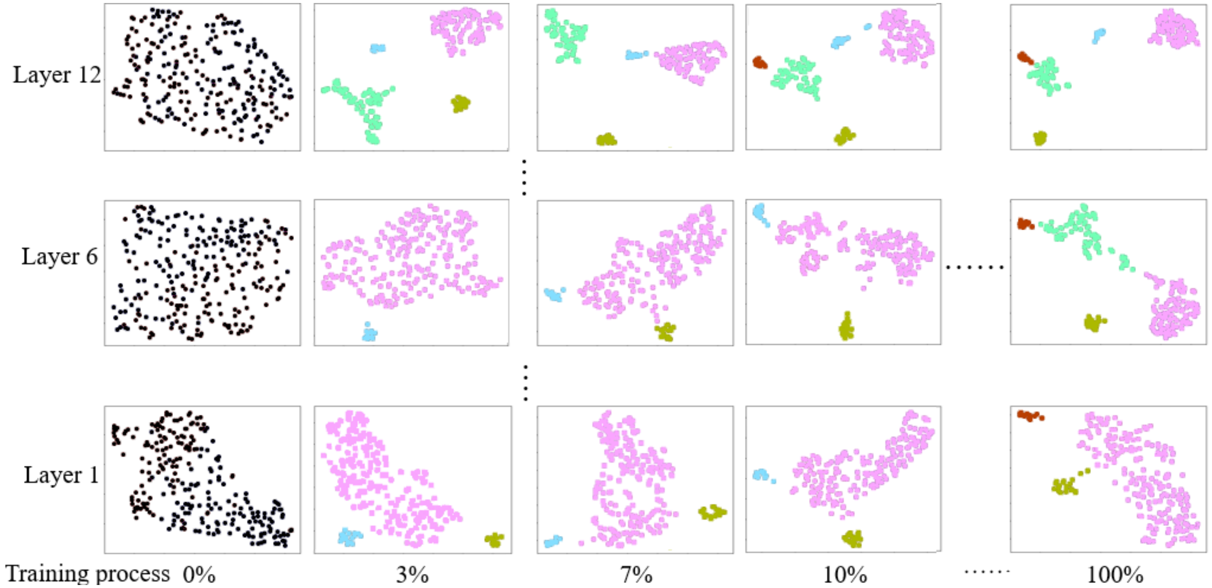
Method



- Cluster-guided Sparse Expert:
  - **Initialization**: train a baseline dense model devoid of any expert structure.
  - **Low-Dimension Clustering**: use a Gaussian random initialized matrix to project embeddings to a low-dimensional space, and perform clustering algorithm in all layers.
  - **Select Layer**: introduce MoE on layers with larger cluster distance-radii ratio. Expert number is equal to the cluster number.
  - **Dispatch**: dispatch new data to its nearest cluster $i = \arg\min_{j=1}^{n} ||v' - c_j||/r_j$
  - **Update Clusters**: update center $c_i^{t+1} = \alpha \cdot c_i^t + (1-\alpha) \cdot v'$

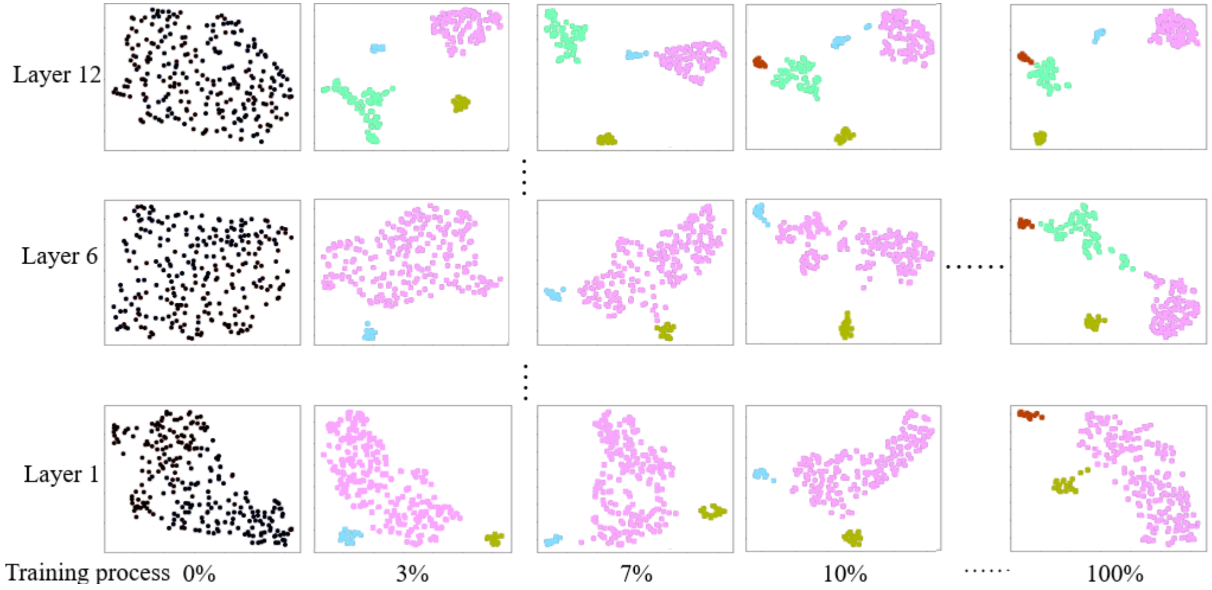
Method

- Representation Cluster Structure
  - Emerge at early stage in training



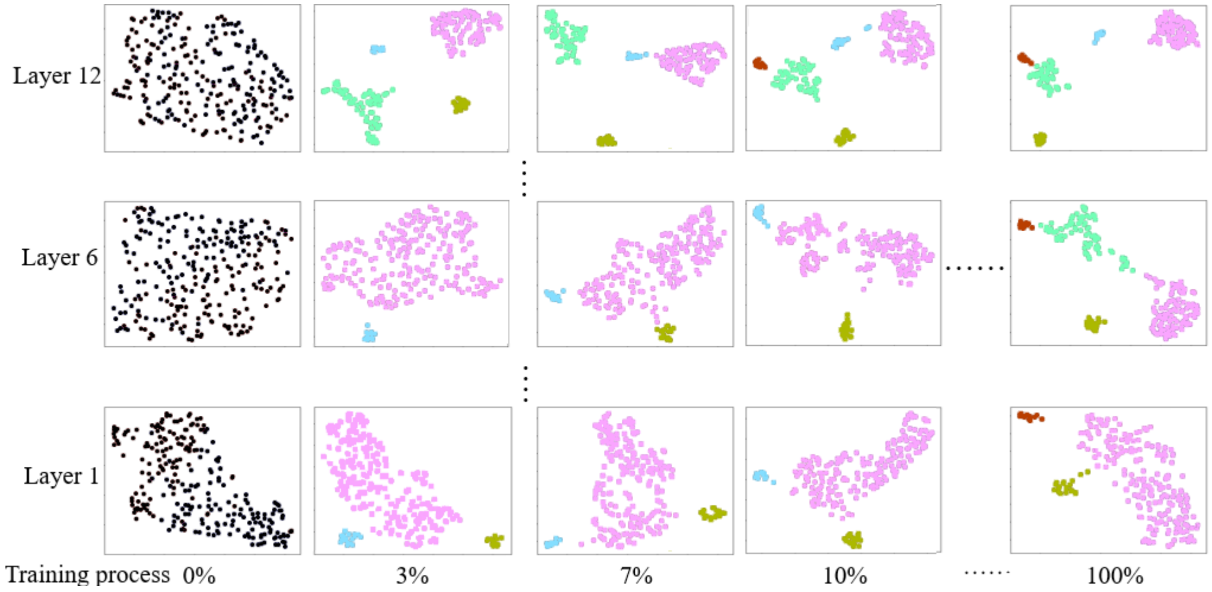
Method

- Representation Cluster Structure
  - Emerge at early stage in training
  - New small outlier cluster may emerge as training progress.



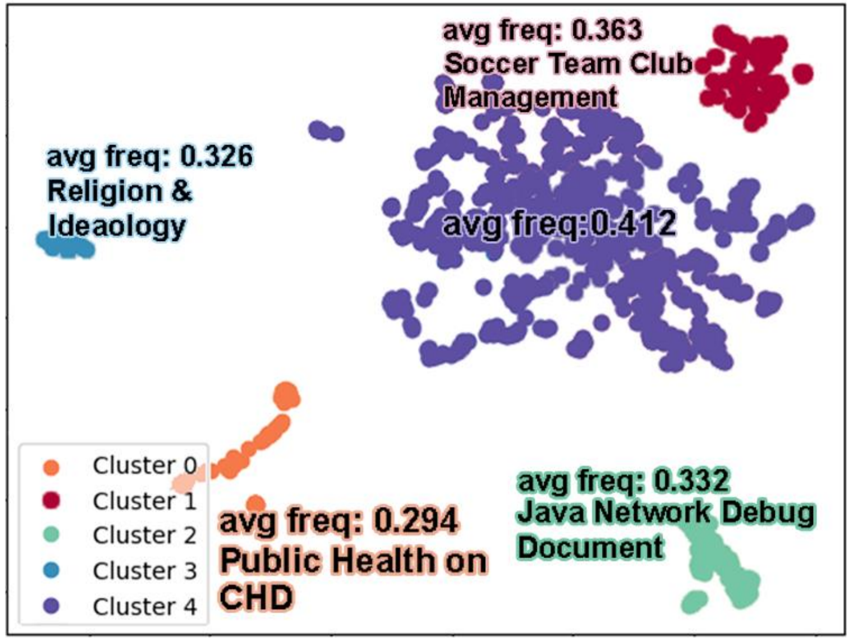
Method

- Representation Cluster Structure
  - Emerge at early stage in training
  - New small outlier cluster may emerge as training progress.
  - More clusters in deep layers than shallow layers.



Method

- Representation Cluster Structure
  - Emerge at early stage in training
  - New small outlier cluster may emerge as training progress.
  - More clusters in deep layers than shallow layers
  - Long-tail data form small, outlier clusters.



Experiments

- Experiment Settings
  - Ours: Undergoes a pretrained phase, reading long-tail domain-specific data once.
  - Baselines: Pretrained on the same dataset and then continue-pretrained on domain-specific datasets.
  - Metrics： Accuracy on downstream domain-specific tasks.

Experiments

- Results-BERT(110M)

Table 1: Results of strategies applied on BERT

Models	Pretrain ppl	Overruling	Casehold	GAD	EUADR	SST2	average
BERT/med	37.00	86.67	50.51	67.09	84.23	66.86	71.07 ± 0.22
BERT/legal	37.00	86.67	50.93	66.83	84.79	65.14	70.87 ± 0.23
MoE/med	31.00	85.00	50.49	64.52	83.10	64.79	69.58 ± 0.20
MoE/legal	31.00	85.83	50.30	64.32	84.79	63.88	69.82 ± 0.19
Ours/MoA	28.25	86.62	**50.94**	**72.90**	90.09	66.60	73.43 ± 0.18
Ours/MoF	34.64	**89.10**	50.82	71.65	**91.23**	**67.98**	**74.16 ± 0.20**

BERT/med exhibited a severe forgetting issue and details will be discussed in the Appendix A.

Experiments

- Results-GPT(130M)

Table 2: Results of strategies applied on GPT

Models	Pretrain ppl	Overruling	Casehold	GAD	EUADR	SST2	average
GPT/med	55.59	88.33	49.82	71.56	84.23	73.90	73.57 ± 0.17
GPT/legal	55.59	89.17	50.58	71.69	81.69	74.50	73.53 ± 0.23
MoE/med	40.69	91.25	50.11	<u>72.77</u>	83.38	72.03	73.91 ± 0.12
MoE/legal	40.69	91.60	49.68	72.66	83.38	71.97	73.86 ± 0.23
Ours/MoA	42.99	<u>91.68</u>	<u>50.70</u>	71.75	**85.91**	<u>74.61</u>	<u>74.93 ± 0.08</u>
Ours/MoF	43.38	**93.33**	**51.26**	**73.30**	<u>85.63</u>	**76.00**	$\mathbf{75.90 \pm 0.19}$

Experiments

- Results-GPT(330M)

Our method also works well as model scaling up.

Table 3: Results of strategies applied on 330M GPT

Domain	Task	GPT/tuned	MoE/tuned	CSE/ w/o tune
academic	chem-prot	36.25	36.25	36.25
academic	MAG	63.22	64.91	**65.47**
academic	rct-20k	76.95	78.28	**80.15**
environment	clim. det.	78.94	79.90	**80.26**
environment	clim. sent.	66.81	68.31	**69.98**
financial	FPB	16.83	25.00	**40.11**

Experiments

- Results-GPT(330M)

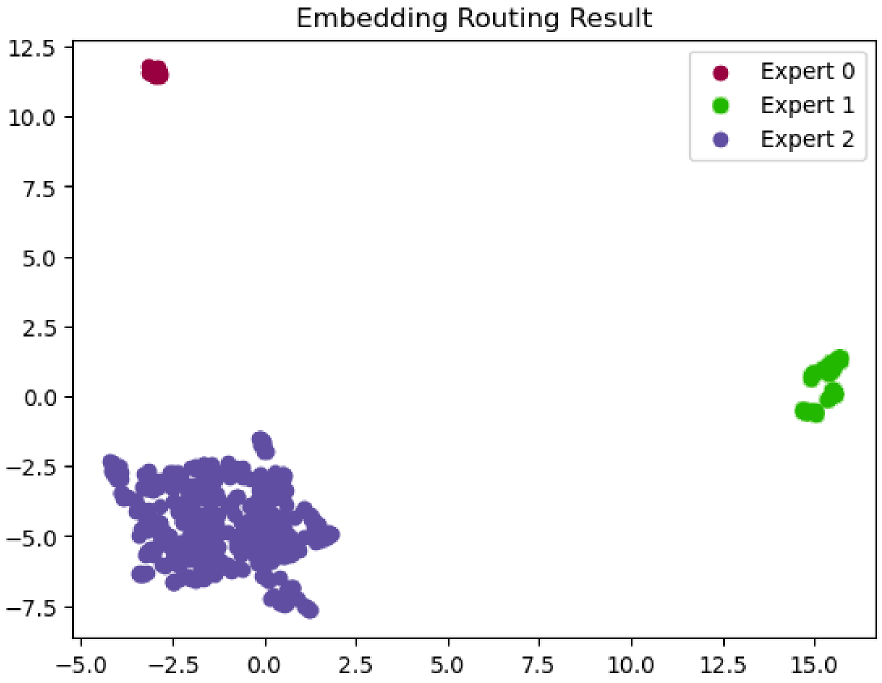
Our method learn long-tail domain knowledge without hurting the performance of general tasks.

Table 8: Results of general tasks tested on GPT 330M trained with 20B tokens

Task	Domain	Freq. Score	Baseline(tuned)	MoE(tuned)	Ours(w/o tune)
COLA	general	0.389	69.10	69.10	**69.20**
QNLI	general	0.325	**60.17**	60.06	59.72
MRPC	general	0.343	70.18	71.75	**71.98**
QQP	general	0.380	73.28	74.47	**75.95**
SST2	general	0.327	74.50	72.03	**76.00**
average	general	-	69.45(-1.12)	69.48(-1.09)	**70.57**

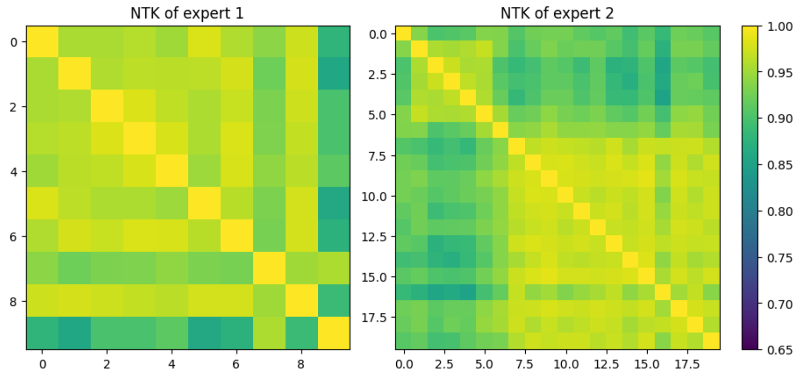
Experiments

- Representation Space Analysis
  - Cluster-guided correct dispatching

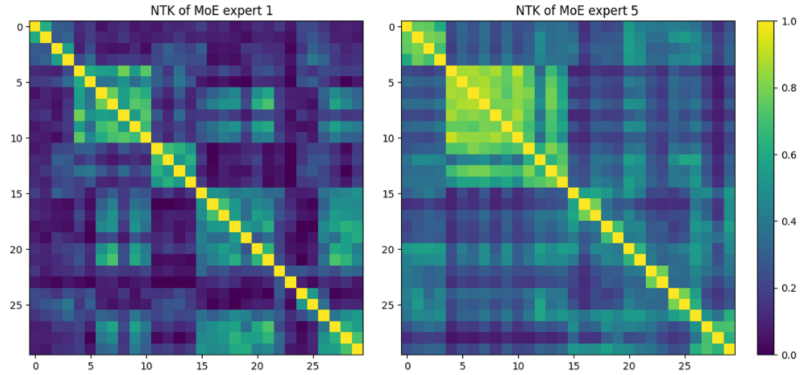


Experiments

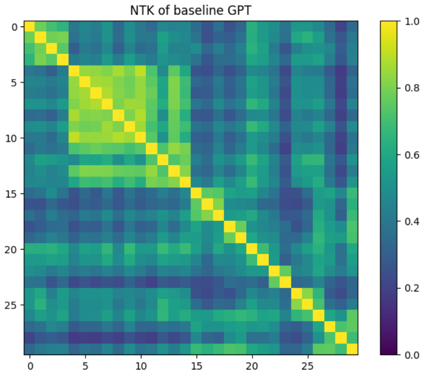
- Representation Space Analysis
  - Cluster-guided correct dispatching
  - Higher gradient consistency on each expert



Ours



MoE



baseline

Thank you!