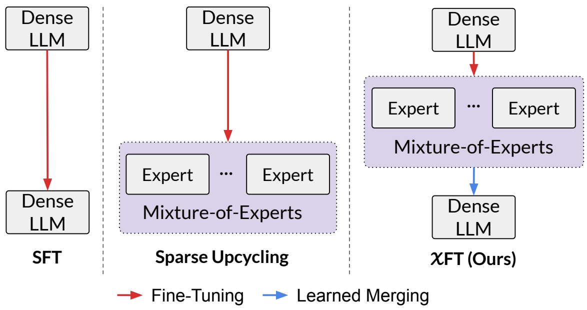
{yifeng6, lingming}@illinois.edu

Abstract



//github.com/ise-uiuc/xft.

ation (Chen et al., 2021; Austin et al., 2021), code

engineering (Lai et al., 2022).

MoE (Dai et al., 2024) and MoCLE (Gou et al., 2024). $\mathcal{X}$FT also includes a novel routing weight

expert and the original dense layer, which will otherwise lead to performance degradation (Wu et al.,

man et al., 2022), $\mathcal{X}$FT uses a learnable model

pass@1 on HumanEval and 64.6 pass@1 on Hu-

Mixture-of-Experts (MoE) can efficiently scale up

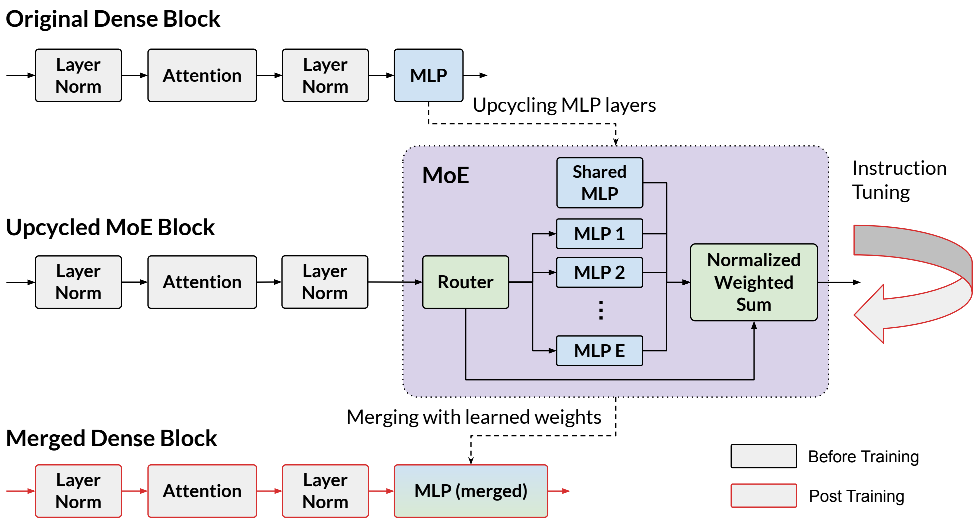
$$\sum^{N}(g_{i,t}\mathrm{FFN}_i(\mathbf{u}_t^l)) + \mathbf{u}_t^l$$

$$\begin{cases} s_{i,t} & s_{i,t} \in \mathrm{Topk}(s_t, K) \\ & \end{cases}$$

efficient fine-tuning (Chen et al., 2022).

et al., 2020; Du et al., 2022; Fedus et al., 2022;

RAMoE (Dou et al., 2023) and MoCLE (Gou et al.,



$$\sum^{N}$$

$$\begin{cases} 1 - s_{t\max} & i = 1 \\ \text{Softmax}_i(s_{i,t}) \cdot s_{t\max} & s_{i,t} \in S_{tK} \end{cases}$$

$$\Big\{ \text{Softmax}_i(\mathbf{u}_t^l$$

$$\sum^{N} \alpha_i^l W_i^l \quad (3)$$

$$\arg\min_{\alpha} \sum^{m} \mathcal{L}(f(x_j; \theta_o, (\sum^{N} \alpha_i^l W_i^l)_{1:L}), y_i) \quad (4)$$

works (Dai et al., 2024; Gou et al., 2024), we

$$\sum^{N} \alpha_i^l W_i^l \quad (5)$$

$$\arg\min_{\alpha} \sum^{m} \mathcal{L}(f(x_j; \theta_o, \overline{W^l}_{1:L}), y_i) \quad (6)$$

LLMs. $\mathcal{X}$FT achieves 67.1 pass@1 on HumanEval

it the new state-of-the-art small code LLM (<3B).

Evol-Instruct (Luo et al., 2023) dataset contain-

ing DeepSeek-Coder-Base 1.3B, DeepSeek-Coder-Instruct 1.3B (Guo et al., 2024), Phi-2 2.7B, and STABLE-CODE 3B (Pinnaparaju et al., 2024).

HumanEval (Chen et al., 2021) and MBPP (Austin

				HumanEval (+)	MBPP (+)
GPT-3.5 (May 2023)	-	Private	-	73.2 (66.5)	-
STABLE-CODE	3B	-	-	28.7 (25.6)	53.6 (44.1)
DeepSeek-Coder-Base	1.3B	-	-	28.7 (25.6)	55.6 (46.9)
Phi-2	2.7B	-	-	48.8 (45.1)	62.7 (52.9)
DeepSeek-Coder-Instruct	1.3B	Private	2B	65.2 (59.8)	63.9 (53.1)
SFT_{DS}	1.3B	Evol-Instruct	0.3B	61.6 (57.3)	59.6 (49.1)
EWA_{DS}	1.3B	Evol-Instruct	0.3B	**67.1** (63.4)	58.9 (48.4)
MoE_{DS}	8×1.3B	Evol-Instruct	0.3B	65.2 (62.2)	60.4 (50.1)
$\mathcal{X}\text{FT}_{\text{DS}}$	1.3B	Evol-Instruct	0.3B	**67.1** (**64.6**)	**60.4** (**50.1**)

Table 1: Pass@1 results of different LLMs on HumanEval (+) and MBPP (+) computed with greedy decoding, following the setting of prior works (Wei et al., 2023; Liu et al., 2023). We report the results consistently from the EvalPlus (Liu et al., 2023) Leaderboard. Note that numbers in bold refer to the highest scores among all 1.3B

		C++	PHP	Java	JS	Swift	Rust	
DeepSeek-Coder-Base	1.3B	28.1	22.9	27.2	28.7	10.9	18.0	22.6
SFT_{DS}	1.3B	40.4	38.5	**40.2**	46.2	16.4	27.7	34.9
EWA_{DS}	1.3B	39.4	38.4	37.3	45.2	20.9	28.6	35.0
$\mathcal{X}\text{FT}_{\text{DS}}$	1.3B	**42.7**	**41.5**	36.0	**49.7**	**25.3**	**32.1**	**37.9**

prior works (Wei et al., 2023; Luo et al., 2023): `temperature` $= 0.2$, `top_p` $= 0.95$, `max_length` $= 512$, and `num_samples` $= 50$. All models are evaluated using `bigcode-evaluation-harness` (Ben Allal et al., 2022).

the sparse upcycling (Komatsuzaki et al., 2023) baseline that does not employ any shared expert.

in most recent works (Dai et al., 2024; Gou et al.,

with (1) directly merging experts with initialized mixing coefficients, and (2) the learnable merging

is the same setting as the learned soup in Model Soups (Wortsman et al., 2022) and is described in Eq. (3) and Eq. (4). Specifically, we initialize the

0.75 and that of the other 7 normal experts as $\frac{1}{28}$ for fair comparison. As shown in Table 5, trained

ing coefficients for merging. Furthermore, removing the shared rate setting will largely degrade the

		np	pd	plt	py	scp	tf	sk	
DeepSeek-Coder-Base	1.3B	25.1	5.8	34.5	12.7	9.8	11.1	12.7	16.4
SFT_{DS}	1.3B	30.9	17.0	40.5	32.7	18.3	21.1	24.4	25.9
EWA_{DS}	1.3B	32.9	19.4	**41.8**	25.7	17.7	**22.2**	33.0	27.8
MoE_{DS}	8×1.3B	33.2	21.3	38.4	41.8	21.8	23.5	37.5	30.0
$\mathcal{X}\text{FT}_{\text{DS}}$	1.3B	**32.9**	**20.2**	38.9	**41.4**	**21.1**	16.9	**37.5**	**29.3**

Table 3: Pass@1 results on DS-1000 (completion format) with `temperature` $= 0.2$, `top_p` $= 0.5$, `max_length` $=$

MoE_{DS}	**65.2**	**62.2**
MoE		
MoE		

1.00	63.4	60.4

$\mathcal{X}\text{FT}_{\text{DS}}$ (INIT)	66.5	64.0
- Shared Expert Rate	66.5	64.0

$\mathcal{X}$FT$_{\text{STABLE}}$	**68.3**	**62.2**

SFT$_{\text{DS}}$
SFT$_{\text{DS}}$

$\mathrm{SFT_{TL}}$	**25.38**	23.30	24.20	26.78	24.97
$\mathrm{MoE_{TL}}$	23.85	26.32	27.40	28.03	26.11
$\mathcal{X}\mathrm{FT_{TL}}$	23.91	**26.49**	**27.72**	**28.29**	**26.30**

References

els. https://github.com/bigcode-project/bigcode-evaluation-harness.

//github.com/sahil280114/codealpaca.

alignment.

mixture-of-experts.

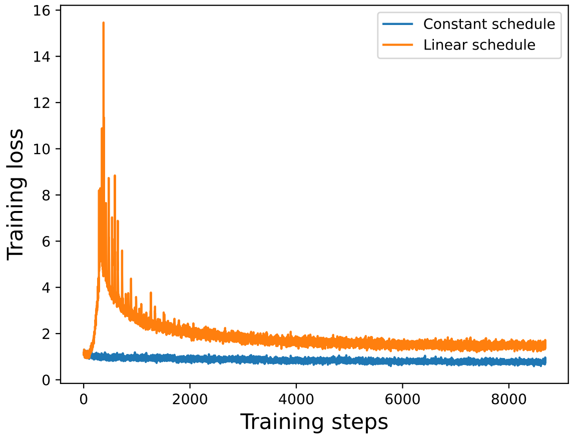
training.

standing.

instruct.

Mixture-of-Experts"

tasks.



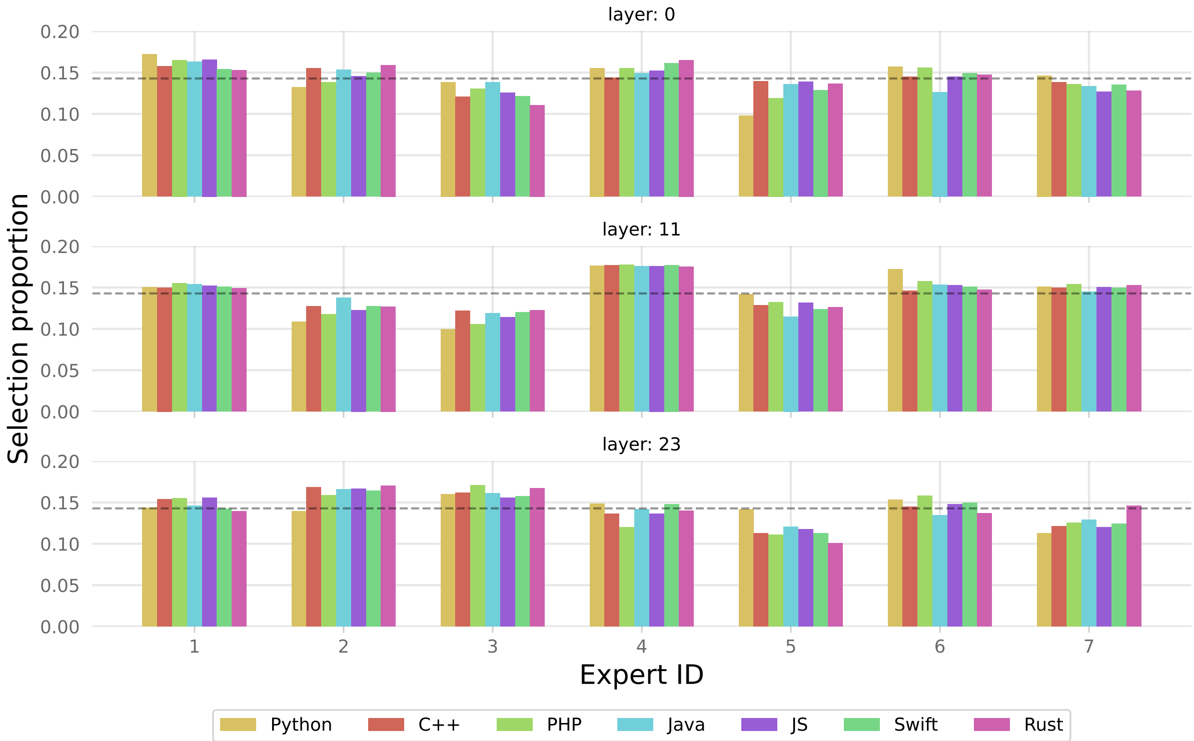
$\mathcal{X}\text{FT}_{\text{DS}}$ vs. EWA_{DS}	2.6e-18	8.0e-23
$\mathcal{X}\text{FT}_{\text{DS}}$ vs. SFT_{DS}	9.6e-30	3.7e-33

EWA_{DS}	62.7	58.8

on HumanEval (+) computed with sampling. $\mathcal{X}$FT

works (Jiang et al., 2024; Xue et al., 2024). 1681

et al., 2021).



always assigned to it. The gray vertical line marks

connected with one feed-forward network (FFN)

attention layer connected with an MoE layer.

as $(1-\alpha)\mathbf{e}_1(\mathbf{u}_t)+\alpha\mathbf{e}_2(\mathbf{u}_t)$, where $1-\alpha$ is

In this simplified scenario, if we denote $f(x;\theta)$

$(1-\alpha)f(x;\theta_1)+\alpha f(x;\theta_2)$! Consequently, the